NASA Contractor Report 189652, Volume I

An Installed Nacelle Design Code Using a Multiblock Euler Solver

Volume I: Theory Document

H. C. Chen

Boeing Commercial Airplane Group
Seattle, Washington

Contract NAS1-18703
September 1992

National Aeronautics and
Space Administration

Langley Research Center
Hampton, Virginia 23665-5225

TABLE OF CONTENTS

1.0 Summary

An efficient multiblock Euler design code was developed for designing a nacelle installed on geometrically complex airplane configurations. This approach employed a design driver based on a direct iterative surface curvature method developed at NASA-Langley. A general multiblock Euler flow solver was used for computing flow around complex geometries. The flow solver used a finite-volume formulation with explicit time-stepping to solve the Euler equations. It used a multiblock version of the multigrid method to accelerate the convergence of the calculations. The design driver successively updated the surface geometry to reduce the difference between the computed and target pressure distributions. In the flow solver, the change in surface geometry was simulated by applying surface transpiration boundary conditions to avoid repeated grid generation during design iterations. Smoothness of the designed surface was ensured by alternate application of streamwise and circumferential smoothings. The capability and efficiency of the code was demonstrated through the design of both an isolated nacelle and an installed nacelle at various flow conditions. Information on the execution of the computer program is provided in Volume II which is titled "An Installed Nacelle Design Code Using a Multiblock Euler Solver, Vol. II: User Guide".

2.0 Introduction

It has been roughly 40 years since the first jet transport flew. Different models have been built by various manufacturers, and, although each manufacturer has their own design and manufacturing approach, there are areas of common concern. One such area is the engine installation. Historically, the aerodynamic design of the engine nacelles and struts were done without considering the interactions between the wing and the strut/nacelle. Some modern designs have the nacelle/strut pitched and/or toed to line up the inlet with the local flow. Still, the common practice is to design the wing without the nacelle/strut and to design the nacelle as an isolated object. The designers who integrated the propulsion system into the airframe had very little say in modifying either the wing geometry or the nacelle geometry in order to avoid significant performance degradation from interference between the wing and the nacelle/strut.

NASA-Langley launched an effort to look into the nacelle installation effect. Both computational work and wind tunnel tests were carried out to gain insight on the interference between the wing and the nacelle/strut (Ref. 1). This report documents the development of a CFD code that would allow the designer to reshape a nacelle installed on a wing/body/strut/nacelle configuration.

There are two major challenges in the detailed aerodynamic design of an installed nacelle. Firstly, it requires a CFD code capable of dealing with an extremely complex geometry as well as proper representation of the physics. Secondly, design of the nacelle surface must be performed under the rigid constraint of fixed leading and trailing edges.

In the beginning of the present study, we acquired NASA-Langley's direct iterative surface curvature (DISC) design driver (Ref. 2) and the general multiblock Euler (GMBE) solver (Refs. 3, 4, 5) as the technology basis. The design driver provides a means for successive updates of the nacelle surface geometry. The GMBE solver provides a flow solution for complex geometry with proper representation of the physics except for viscous effects. In GMBE, the capability to accept surface transpiration boundary conditions has been developed

to simulate the effect of surface change during design iterations. Re-gridding, to verify the accuracy of representing surface change using the transpiration boundary condition, is only done once for the final analysis.

The second challenge in designing an installed nacelle is related to the problem of geometric closure. The original DISC design driver relates the geometry change to the difference between the computed and the target pressure distributions for each streamwise strip on the nacelle. The design driver first re-defines a new strip in an unconstrained manner. The problem of geometry closure at the point farthest downstream (i. e., usually the trailing edge) of a design segment is resolved by rotating the segment hinged at a point farthest upstream (i. e., usually the leading edge) of the segment.

In the present study, it was found that, for some test cases, the amount of rotation required could become large during the initial design iterations. This could affect the convergence of the iterative process. For this reason, a procedure was developed to reduce the required rotation. The relationship between geometry change and the difference in pressure distributions was modified before surface redefinition. Numerical issues in implementing such a procedure have been addressed.

Surface smoothness is a critical requirement in the design of a three dimensional aerodynamic surface, such as that of an installed nacelle. In the present study, surface smoothness was ensured by applying streamwise and circumferential smoothings for each design iteration.

There were two test cases in the study. For both test cases, design iterations followed pre-design analyses. The first case was a two-block isolated nacelle design problem. The design code was able to recover the target nacelle from the corresponding target pressure at a supercritical flow condition. The second case designed an installed nacelle on a generic twin-engined, NASA low-wing transport configuration (Ref. 6). Results from the two test cases demonstrated the accuracy and efficiency of the design code. The cost for a design was comparable to that of the pre-design analysis. A user manual for the installed nacelle design code is given in Vol. II of the present report (Ref. 7).

3.0 Methodology

The direct iterative surface curvature (DISC) method developed by Smith and Campbell (Ref. 2) was adapted to design the external fan cowl of an installed nacelle. The basic theory has been described previously (Ref. 2). Salient points of the theory are discussed here. Its implementation and new findings are then presented.

3.1 Theory

The algorithm developed by Smith and Campbell begins with the inviscid momentum equation. For 2-D flows, the equation can be written in the streamline curvature form,

$$dq/q = -\kappa(n)dn \tag{1}$$

where q is the local speed, κ is the streamline curvature, and n is the distance normal to the streamlines. Following the analysis of Barger and Brooks (Ref. 8), the streamline curvature decays exponentially with respect to the distance normal to the airfoil surface,

$$\kappa(n) = \kappa_0 e^{-Cn} \tag{2}$$

where C is a positive constant and the subscript 0 denotes the value on the airfoil surface. Substituting this expression into the momentum equation and integrating gives

$$\kappa_0 = C \ln(q_0/q_\infty) \tag{3}$$

where the subscript ∞ indicates the freestream value. If the change in speed Δq_0 is small, the Taylor series expansion of κ_0 about q_0 can be performed and higher order terms of Δq_0 can be dropped. The result is

$$\Delta\kappa_0 = C\, \Delta q_0/q_0 \tag{4}$$

Letting C be proportional to κ_0, as suggested in Reference 8, leads to

$$\Delta\kappa = A_1 \kappa \Delta q/q \tag{5}$$

where A_1 is a constant, and subscripts 0 have been dropped for simplicity. If the perturbations caused by the airfoil are assumed to be small, it follows that: (a) the denominator q can be expressed as q_∞, (b) the numerator Δq can be expressed as Δu, where u is the perturbation velocity in the freestream direction, and (c) the pressure coefficient C_p can be expressed as $-2u/q_\infty$. Equation (5) then becomes

$$\Delta\kappa = A\kappa\Delta C_p \tag{6}$$

where $A=-A_1/2$ and ΔC_p is the change in the pressure coefficient. As the curvature κ approaches zero, a small change in curvature would result in a very large change in the pressure coefficient C_p. This situation, however, does not actually occur in real flows. It is the result of the assumption of Equation (2) used in integrating the momentum equation. To circumvent this singular behavior, Equation (6) is modified to read

$$\Delta\kappa = A(1 + \kappa^2)^B \Delta C_p \tag{7}$$

where B is a constant ranging from 0.0 to 0.5 and constant A has different signs depending on whether we are on the upper surface or on the lower surface. The term $(1 + \kappa^2)^B$ approaches unity for small values of curvature κ. It will approach κ^{2B} for large values of curvature κ and the resulting κ^{2B} will be large for B = 0.5. Consequently, a small change in C_p may lead to a large change in curvature. High curvature regions usually occur at the leading edge of an airfoil or nacelle section. In such regions, if the flow is supercritical, the large change in curvature may affect the stability of the design iterations. Using a low value for the constant B can stablize the iterative procedure.

The final assumption is that the change in curvature results in only small changes in the local slope so that the relationship between the curvature and the second derivative can be written as

$$\Delta y'' = \Delta\kappa(1 + y'^2)^{1.5} \tag{8}$$

Integrating the above equation twice will yield Δy. Equations (7) and (8) provide the relationship between the change in surface pressure coefficient C_p and the local second derivative y" in a locally subsonic flow. This relationship seems to work for both subsonic and transonic flow cases, although the theory applies only to locally subsonic flow. However, the convergence was found to be rather slow for flows with local Mach number greater than 1.15. As discussed in Reference 9, the relationship between the surface pressure coefficient C_p and the surface geometry changes character as the flow becomes locally supersonic. The development of a hybrid algorithm, similar in concept to that in Reference 9, was therefore based on supersonic thin airfoil theory. Surface pressure coefficient C_p in a supersonic inviscid flow is related to the local slope y′ by

$$C_p = 2y' / [(M_\infty)^2 - 1]^{1/2} = C_1\, y' \tag{9}$$

This equation can be rearranged to yield

$$\Delta y' = C\Delta C_p \tag{10}$$

Differentiating with respect to x gives

$$\Delta y'' = C\, d(\Delta C_p)/dx \tag{11}$$

Strictly speaking, the above theory is only valid for supersonic freestream Mach numbers. Therefore, for transonic flows, the value for the "constant" C is left to be determined empirically. If the freestream Mach number in Equation (9) is replaced with a so-called "switching" (Ref. 2) Mach number of 1.15, the value of C would be 0.28. A switching Mach number is defined as a cut-off number such that whenever the local Mach number exceeds the cut-off value, then Equation (11) instead of Equation (8) is employed. Our experience showed that good results can be obtained by using a value of 0.16 for the constant C which under-relaxes the changes in geometry and therefore stabilizes the process of successive iteration. The corresponding switching Mach number is 1.05. This leads to a "hybrid" algorithm.

3.2 Implementation

The algorithm presented in Section 3.1 was initially developed for the study of 2-D airfoils. It has also been applied to wings (Ref. 10) and isolated nacelles (Ref. 11). Ideally, this method should use constant azimuthal cuts of the nacelle. In actual practice, however, not all grid lines follow the constant azimuthal cuts, due to the presence of the strut. In this report, the method is applied to streamwise grid lines of an installed nacelle to redesign the nacelle.

As is the case for the study of wings and isolated nacelles, the analysis of a baseline configuration (pre-design analysis) needs to be performed. The analysis identifies the areas requiring redesign because of undesirable flow features. Strong shocks and severe pressure recoveries, that might cause boundary layer separation, are a few examples of such features. Desired pressure distributions, also referred to as target pressure distributions, are then input to the code. The term ΔC_p is defined as the difference between the target pressure distribution and the pressure distribution from the pre-design analysis. The code uses equations (8) and (11) to convert ΔC_p into changes in second derivatives of a streamwise strip of the local surface geometry. Integrating the changes in the second derivative in the streamwise direction yields the new geometry. The new geometry is then re-analyzed and the difference between the resulting surface pressure and the target pressure is used to compute the successive changes in surface geometry second derivatives. The process repeats until the number of geometry updates reaches a user-specified number. The evolution of the nacelle surface C_p distributions and geometries are stored during this process. The C_p distributions and geometries are then displayed and compared between successive design iterations to check for convergence. If the design iteration is insufficient to achieve convergence, then a continuation run is made to further the design iterations. In the case where the target C_p distribution is unrealistic, the computed C_p distributions may settle down at a level that is different from the target rather than "converge" to the target distribution.

The geometry updates change the surface grid locations. Ideally, a new field grid should be generated, but since grid generation is independent of both the flow

solver and the geometry change module, re-gridding during the iterative process can be a tedious task. Instead, a surface transpiration model, similar to the one used in the boundary layer displacement thickness simulation (Ref. 12), is used to simulate the change in surface geometry. A simple flux balance (Fig. 1) shows that the surface transpiration velocity q_n is related to the surface geometry change by

$$q_n = 2\,[(\Delta y \rho u)_2 - (\Delta y \rho u)_1]\,/\,[(\rho_1+\rho_2)\,l_{12}] \tag{12}$$

where $\Delta y = y_{new} - y_{old}$, ρ is the density, u is the velocity component in the x direction, and l_{12} is the arc length on the old surface between stations 1 and 2. Another option is to use the transpiration mass flux obtained from a similar equation

$$(\rho q)_n = [(\Delta y \rho u)_2 - (\Delta y \rho u)_1]\,/\,l_{12} \tag{13}$$

The surface transpiration velocity has been used to simulate the geometry changes for all design iterations except the final converged solution in which the volume grid around the perturbed geometry is regenerated using a grid perturbation method. The original volume gird is perturbed such that the surface change introduced by Equations (8) and (11) propagates into the flow field. The newly designed surface grid is on one face of the block, while the geometry on the opposite face remains fixed. A blending function is used between the faces to allow the change in surface grid to decay smoothly to zero on the opposite face. Finally, an analysis on the new grid verifies the accuracy of representing the geometry changes using the transpiration boundary condition.

There are two basic requirements for the grid perturbation method to work. The first one is that the surface change does not alter the layout of the field blocks. The second one is that the external space is divisible into two regions (Fig. 2a) such that only the grids within a few blocks in one of the regions requires redistribution. In addition, the surface change should be moderate in magnitude and the first few layers of grid cells near the surface should have sufficient spacing to prevent grid line cross-over.

Figure 2b shows the front view of a cross-stream cut of region-1 with the nacelle and the block boundaries. This region is divided into five blocks, BLK_1 to BLK_5. The design process produces a new surface and moves the four edges to new positions A′, B′, C′ and D′. BLK_2 can be re-gridded easily from B-A-5-6 to B′-A′-5-6. The re-gridding of BLK_1 is slightly more involved. It is first divided into two sub-blocks, 1-2-3-A and A-3-4-5. Each sub-block can then be re-gridded in a manner similar to BLK_2. The important part is that along edge 3-A′ both sub-blocks must use the same points. Similarly BLK_5 must be divided into two sub-blocks, while BLK_3 must be divided into three sub-blocks before re-gridding.

In some cases, the block layout for the flow field around a transport type of wing/body/strut/nacelle configuration can result in points where special treatments are required. Figure 3 shows two types of points. The one in Figure 3a does not need the special treatment and is called a regular point. The one in Figure 3b needs the special treatment and is called a singular or fictitious point. The pressure coefficient at the fictitious point is not uniquely defined unless special care is taken. The multiblock Euler flow solver stores an extra layer of data beyond the block boundary. The pressure along the fictitious point in BLK_1 of Figure 3b is calculated by averaging the cell center values from cells 1, 2, 3, and 4. The pressure along the same point for BLK_2, however, is computed by averaging the cell center values from cells 2, 3, 4, and 5. This leads to possible non-uniqueness of pressure at the fictitious point. This does not affect the accuracy of the analysis results since the pressure at the cell vertex is not used in the flux balance. However, for the design code, where the surface pressures are compared with the target pressure to compute the geometry change, the non-uniqueness of the pressure at the fictitious point is a real problem. One solution to this problem is to reorganize the data along the nacelle outer surface into circumferential rings (Fig. 4). The circumferential data is then used for pressure averaging along the fictitious point thus guaranteeing uniqueness.

The flow field around a transport type wing/body/strut/nacelle configuration may be subdivided into many blocks. The design code goes through the flow analysis part for all blocks although only a few blocks involve the updated geometry. An option, called local design, was developed for the code to apply the flow solver less frequently to those blocks that are not in the immediate

neighborhood of geometry changes. The code will, therefore, spend relatively more time in an area where the flow parameters are most likely to be affected by the geometry change. This procedure should improve the efficiency of the code.

3.3 Findings

The basic geometry alteration algorithm DISC was supplied by NASA-Langley. Problems occurred during adaptations of DISC for design code development and study of the problems resulted in several findings. These findings were mostly related to geometry issues such as the closure of the newly designed section, the stability of the design iterations, and the surface smoothness. They are presented below.

3.3.1 Geometry Closure

Most likely, the new geometry obtained by integrating Equation (8) or (11) would not match the original surface at the downstream end of the design region because there are no closure conditions imposed in specifying the target pressure distribution. The original work by Smith and Campbell (Ref. 2) rotates the entire newly designed 'segment' about the upstream end of the design region to force the closure. A 'segment' is the new geometry from the starting location to the end of the design region (e. g., trailing edge point). This type of rigid body rotation about the upstream end of the design region (e. g., leading edge point) is simple and straight forward. It works well if the amount of rotation needed to close the section is small and monotonically decreases during design iterations. For some test cases, however, the amount of rotation required may become large during the early stage of design iteration. In such instances, discontinuities in slope and curvature are introduced at the center of rotation. This may affect the convergence of the iterative process. A procedure to reduce the amount of rigid body rotation in the geometry closure is given in Appendix A.

3.3.2 Surface Smoothing

The new section designed by integrating Equations (8) and (11) may or may not be aerodynamically smooth. Smith and Campbell use a 7-point least-squares curve

fit to smooth the section in the streamwise direction. For 3-D applications smoothing is also needed in the circumferential direction in the case of installed nacelles.

The least-squares curve fit uses a third order polynomial which has two forms, depending on whether the point being smoothed is close to the leading edge or not. Away from the leading edge the polynomial takes the form of

$$y = c_1 + c_2 x + c_3 x^2 + c_4 x^3 \tag{14}$$

For the first 20% of the airfoil chord an additional square root term is added, thus

$$y = c_1 + c_2 x + c_3 x^2 + c_4 x^3 + c_5 x^{0.5} \tag{15}$$

This square root term gives infinity slope at the leading edge x=0. A detailed study of the smoothing procedure based on Equation (14), is given in Appendix B.1.

Installed nacelles designed by the current technique and smoothed by Equations (14) and (15) should have smooth streamwise sections. However, the circumferential grid lines may not be smooth because each streamwise section is designed independently. In addition, the streamwise smoothing using Equations (14) and (15) could also generate some circumferential noise in the geometry. This situation is corrected by applying streamwise and circumferential smoothings alternately.

The circumferential smoothing is applied to the change in geometry, $\Delta r = r_{new} - r_{old}$, where r is the radial coordinate, rather than the geometry itself. Two smoothing techniques are available to the user. One is the 7-point least-squares technique mentioned above. The other is a method based on averaging the difference Δr of neighboring points. The smoothed Δr is added to r_{old} to generate a smooth r_{new} provided that the starting r_{old} is smooth. Both techniques can smooth out Δr in a few passes. However, smoothing by averaging is significantly faster and has been used in this study. More details can be found in Appendix B.2.

Figure 5 plots the Δr as a function of θ before circumferential smoothing.

Typically, Δr has peaks and valleys. They represent the points of maximum geometry change. It is desirable to smooth out the peaks and valleys without moving them to a different θ location.

Figure 5 also shows the Δr after the smoothing step. This step consisted of three passes of smoothing using the averaging technique. Each pass consisted of one circumferential smoothing and one streamwise smoothing. The peaks and the valleys have been smoothed but their θ locations have not changed.

4.0 Results and Discussions of Test Cases

Two sample cases were constructed to test the design code. The first case was an isolated nacelle and the second was an installed nacelle.

The isolated nacelle study was a good check of the DISC implementation in GMBE. Also, experience gained from it was very helpful in the subsequent study of the installed nacelle. From an application stand-point, the design of a suitable isolated nacelle is the starting point for an installed nacelle design. The isolated nacelle design used the calculated surface C_p distribution of another nacelle as the target to check out the design process.

Study of the installed nacelle was divided into two parts. Part one was similar to the isolated nacelle case except that the known geometry to be recovered was an installed nacelle. Part two used a target C_p distribution that was derived by hand-tailoring the results of a pre-design analysis. In all such design exercises, there are of course no guarantees that a physically meaningful geometry actually exists.

4.1 Isolated Nacelle Design

An axisymmetric isolated nacelle, nacelle-1, was analyzed using the multiblock Euler code to provide the pre-design information. A vertical symmetry plane was used and half of the nacelle and external flow field was solved. The flow field was covered by two blocks of grids (Fig. 6). One block covered the internal flow and the other covered the external flow. The grid size for the internal block was 97 x 13 x 9 in the streamwise, radial and circumferential directions respectively. The grid size for the external block was 97 x 25 x 9. There were 49 points in the streamwise direction on the fan cowl.

This nacelle was analyzed at $M_\infty = 0.85$ and $\alpha = 0.0$ degrees for 500 multigrid steps. The converged surface C_p distribution, which is presented in the Figure 7, indicates a strong shock. The same figure also presents the convergence history of the surface C_p distributions. The main features of the flow, such as the pressure peak and shock wave, were well established at 300 multigrid steps. Comparison of C_p at 300 steps and 500 steps shows that there is only a small difference in C_p

distribution near the foot of the shock.

For the purpose of target C_p generation, a computational grid for a slightly different nacelle, nacelle-2, was generated. The block layout and grid size for each block were the same as that of nacelle-1. The x-coordinates of the grid points on the nacelle surface are also the same between the two nacelles. The DISC method alters the radial coordinate r at each grid point in the design region but leaves the x-coordinate unchanged. Using the same x-coordinates for the nacelle surface grid points ensured that the flow features contained in the target C_p could be properly resolved as the geometry approached the target geometry of nacelle-2. Nacelle-2 was analyzed at the same flow condition as nacelle-1 for 500 multigrid steps. The computed C_p had to be well converged to ensure that the resulting C_p (i. e., the one to be used as the target C_p) corresponded to nacelle-2. Results, illustrated in Figure 8, show that nacelle-2 produced a weaker shock than that of nacelle-1. The comparison of nacelle geometries is also shown.

Nacelle-1 was then used as the initial geometry for the design with the surface C_p from nacelle-2 as the target C_p. The entire external surface was included in the design region. The pre-design results at 500 steps, given in Figure 7, were used to provide the initial C_p to start the design iterations. The initial C_p had to be reasonably converged to capture the basic features of the flow, such as the pressure peak and shock. The initial C_p, however, is not required to be fully converged and the pre-design solutions at 300 steps could have been used to start the design. Within each design cycle, there were 50 multigrid steps to solve the flow so as to provide a new C_p for the updated geometry. After six cycles of design iterations, there was good agreement between the designed surface C_p distribution and that of the target as shown in Figure 9. The same figure also presents the geometry recovery of nacelle-2. The convergence history of the C_p distributions, illustrated in Figure 10, shows that the computed C_p essentially reached that of the target C_p in the fourth cycle of the design iterations. The last two cycles improved the minor differences in C_p distributions near the the foot of the shock. Figure 10 also presents the convergence history of the nacelle surface geometries.

Successful completion of this isolated test case was crucial as it provided general guidelines for the design of the installed nacelle. This information

included typical numbers for multigrid steps required to establish the main features of the flow, the number of design cycles required for convergence to the final design, and the number of multigrid steps required for each geometry update. The cost of a design, as measured by the total number of multigrid steps required after the pre-design analysis, was comparable to that of the pre-design analysis. The pre-design analysis took nine minutes of CPU time on a Cray Y-MP using one CPU while the design iterations took six minutes. This kind of high efficiency is vital if a design code is to be used to solve large and complex problems such as the installed nacelle design. Efficient design codes reduce computer resource requirements and allow for better turn-around time, a crucial element in the design process.

4.2 Installed Nacelle Design

The baseline geometry employed in this study was the NASA-Langley 1/17 scale low-wing transport model, a wing/body/strut/nacelle configuration tested in the 16-ft transonic tunnel. Detailed information about the model can be found in Ref. 6. The surface grid is shown in Figure 11; the entire flow field was divided into 32 blocks with 1,170,000 grid points. On the external surface of the fan cowl, there are 45 points in the streamwise direction and 46 points in the circumferential direction. Study of the installed nacelle design was divided into two parts as mentioned previously. Part one focused on the recovery of a target C_p generated from known geometry. Part two studied the actual design of the installed nacelle based on a user prepared target C_p. An identical block layout and grid size was employed in both parts of the study.

In part one, the target geometry was obtained by perturbing the external surface geometry of the baseline nacelle and the perturbed configuration was analyzed at $M_\infty = 0.77$ and $\alpha = 0.5$ degree for 500 multigrid steps, to produce a target C_p. The baseline geometry was used as the initial geometry to redesign the external fan cowl. Pre-design analysis of the baseline geometry was conducted at the same flow conditions for 500 multigrid steps as well. There were 50 multigrid steps between design iterations to provide the C_p for the geometry updates.

In Figure 12, the initial C_p (i. e., from baseline geometry), the target C_p (i. e.,

from perturbed geometry), and the designed C_p distributions are plotted at four circumferential cuts on the nacelle surface. The designed C_p (6th design cycle) compared very well with the target C_p. Figure 13 compares the initial, target, and designed geometries at the same circumferential locations. Chordwise distributions of r are used to represent the surface geometry, where r is the radial coordinate taking the nacelle centerline as the axis. Recovery of nacelle surface geometry is shown in Figure 13. This part of the installed nacelle study establishes confidence in the design code before it is used in part two; a realistic design case where the final geometry is not known apriori.

In part two, the target C_p was user prepared. The specification of the target pressure distribution depends on the intent of the design. In general, strong shocks are to be avoided. Therefore, for a configuration whose pre-design analysis shows strong shocks, such as nacelle-1 in the isolated nacelle study, the target pressure distribution is relatively easy to specify. For configurations where the problems are subtle, the guidelines for target C_p preparation are not always obvious.

The baseline geometry was again used as the starting geometry. The same pre-design analysis used in part one was used here as well. The target C_p was obtained by hand-tailoring the pre-design analysis results. The guiding principle was to reduce the pressure peak that follows the leading edge expansion. This can be seen by comparing the initial and target C_p at the four circumferential cuts in Figure 14. There is a potential problem with this target C_p. Note that while the target C_p distribution has a reduced leading edge pressure peak, the detailed C_p distribution near the leading edge, where expansion takes place, is identical to the initial. Generally, it is difficult to alter the leading edge pressure peak without modifying the leading edge expansion. The actual geometry required for this target C_p most likely does not exist. Therefore, this is an extreme test case for the design code in the sense that some compromise must be made, by the code, between the conflicting requirements of reducing the pressure peak and maintaining the leading edge expansion.

Interestingly, an understanding of the user's perception on how the design code is to be used, can provide important direction and guidance for the development of the design code. Ordinarily the design code is used to establish a fast design

process. Turn around time is likely to be a critical issue and target C_p would probably be prepared rapidly. The user might be interested in modifying the pre-design analysis results to obtain desirable features of the flow, without having to worry about the conflicting requirements contained in the target C_p. Some users might overlook the fact that there was a conflict in the target C_p. Others might know exactly what they were doing but expect the design code to work out the problems. An example of one such target C_p was illustrated in Figure 14. For this type of target C_p, attempts to apply a geometry closure procedure, that would maintain the leading edge geometry (e. g., using the procedure discussed in sub-section 3.3.1), would leave very little room to modify the pressure peak. The outcome of the design would most likely be unsatisfactory.

The designed C_p is compared with the target C_p in Figure 14. The pressure peak was generally reduced. The leading edge expansion was different than the target value. This appears to be the compromise required to reduce the pressure peak. The target C_p distribution allowed no pressure change near the leading edge. This implied no geometry change would occur in this region, based on Equations (8) and (11), without using rigid body rotation for geometry closure. Rigid body rotation, therefore, was the source for leading edge geometry alteration and C_p change.

Since the designed C_p was unable to reach the target C_p, the evolution of the computed C_p was examined (see Figure 15) along a circumferential cut on the inboard side of the nacelle. The computed C_p settled down in the 6th cycle of design iteration. The evolution of nacelle surface geometry r against design iterations is also shown in Figure 15. Here, the geometry also settled down in the 6th cycle. It would be interesting to determine why the iterative process settled down while ΔC_p was still large, since nonzero ΔC_p should be followed by geometry change based on Equations (8) and (11). The reason might be that such a geometry change is essentially cancelled by the subsequent rigid body rotation for geometry closure. This "limit cycle" process would then allow design iterations to settle down even for the unrealistic target C_p in the present study. The smoothing has very little effect on this limit cycle process.

A new computational grid, based on the designed geometry, was generated for a

final analysis. This verification step had to be performed since the design iterations were unable to fully achieve the target C_p. Figure 16 presents a comparison of the designed C_p and the computed C_p based on the new grid along the four circumferential cuts. The excellent agreement between the two results verifies the accuracy of representing the surface geometry change by using transpiration boundary conditions.

The cost for the installed nacelle design was comparable to that of the pre-design analysis. Running on Cray Y-MP using one CPU, the pre-design analysis consumed 3 hours and 40 minutes of CPU time while the design part took 2 hours and 20 minutes. Since the pay-off of CFD research and development relates directly to the amount of user applications, the value of a design code is greatly enhanced by having high efficiency. User acceptance of the design code is likely when the cost for the design is comparable to that of the analysis.

5.0 Concluding Remarks

A multiblock Euler design code was developed for installed nacelle design employing a design driver provided by NASA Langley. The effect of surface geometry change was simulated by imposing surface transpiration boundary conditions. Re-gridding on the converged design was used for final analysis to verify the accuracy of such simulation. Issues, regarding the design of an installed nacelle under the rigid constraint of fixed leading and trailing edges, were addressed. The capability and efficiency of the design code were demonstrated by designing isolated and installed nacelles. The code was effective in re-designing a nacelle to weaken or eliminate a strong shock, as well as to reduce the leading edge pressure peak. For both test cases, the cost for design was comparable to that of the pre-design analysis. Information regarding the use of this design code is available in Vol. II.

6.0 Acknowledgement

Mainframe computing has been provided by the Numerical Aerodynamic Simulation Complex at NASA Ames Research Center.

7.0 References

1. Naik, D. A., Chen, H. C., Su, T. Y., and Kao, T. J., "Euler Analysis of Turbofan/Superfan Integration for a Transport Aircraft," AGARD Fluid Dynamic Panel Symposium on Engine-Airframe Integration, Fort Worth, TX, October, 1991.

2. Smith, L. A. and Campbell, R. L., "A Method for the Design of Transonic flexible Wings," NASA Technical Paper 3045, December, 1990.

3. Chen, H. C., Su, T. Y., and Kao, T. J., "A General Multiblock Euler Code for Propulsion Integration, Volume I: Theory Document," NASA CR-187484, Volume I, May 1991.

4. Su, T. Y., Appleby, R. A., and Chen, H. C., "A General Multiblock Euler Code for Propulsion Integration, Volume II: User Guide for BCON, Pre-Processor for Grid Generation and GMBE, " NASA CR-187484, Volume II, May 1991.

5. Chen, H. C., "A General Multiblock Euler Code for Propulsion Integration, Volume III: User Guide for the Euler Code," NASA CR-187484, Volume III, May 1991.

6. Pendergraft, Jr., O. C., Ingraldi, A. M., Re, R. J., and Kariya, T. T., "Nacelle/Pylon Interference Study on a 1/17-Scale, Twin-Engine, Low-Wing Transport Model," AIAA 89-2480, July 1989.

7. Chen, H. C., "An Installed Nacelle Design Code using a Multiblock Euler Solver, Volume II: User Guide," NASA CR-189652, Volume II, 1992.

8. Barger, R. L., and Brooks, C. W., Jr., "A Streamline Curvature Method for Design of Supercritical and Subcritical Airfoils," NASA TN D-7770, 1974.

9. Davis, W. H. Jr., "Technique for Developing Design Tools From the Analysis Methods of Computational Aerodynamics," AIAA Paper 79-1529, July 1979.

10. Yu, N. J. and Campbell, R. L., "Transonic Airfoil and Wing Design

Using Navier-Stokes Codes," AIAA Paper 92-2651.

11. Lin W. F., Chen A. W. and Tinoco, E. N., "3D Transonic Nacelle and Winglet Design," AIAA 90-3064-CP, August, 1990.

12. Allmaras, S. R., "A Coupled Euler/Navier-Stokes Algorithm for 2-D Unsteady Transonic Shock/Boundary-Layer Interaction," PhD Thesis, MIT, February 1989.

Appendices

Appendix A

Reduction of Rotation

A procedure to reduce the amount of rigid body rotation in the geometry closure was developed for the current study. It is based on the observation that Equations (7) and (11), which relate surface curvature change to the change of surface pressure coefficient ΔC_p, are approximate in nature. Assume that a geometry exists; one that satisfies the fixed leading and trailing edge constraint, and corresponds to the target C_p distribution. In the early stage of design iteration, the error in the approximation is likely to be large for a large ΔC_p distribution. Such errors can lead to trailing edge mismatch and violate the closure constraint. The required curvature change as estimated by Equations (7) and (11) may be further modified to satisfy the closure constraint. Such a modification, cannot be arbitrary, and should be in qualitative agreement with Equations (7) and (11). The surface geometry change which is needed to deliver the target pressure, according to Equations (7) and (11), has the effect of rotating the local segment by the amount $\Delta\theta_i$ at the grid point i. $\Delta\theta_i$ is computed by integrating Equations (8) and (11) and each $\Delta\theta_i$ is proportional to the ΔC_p. From the starting point of the design region to the end, there are points where the change of local slope $\Delta\theta_i$ is positive and points where they are negative. A positive $\Delta\theta_i$ will raise the trailing edge and a negative one will lower the trailing edge. Ideally, the net effect should be such that the downstream end of the newly designed segment matches the original geometry. In practice the matching is not achieved. In such a case, the distribution of $\Delta\theta_i$ is modified by rescaling the $\Delta\theta_i$ with the following steps.

Assume that there are n_a points with positive $\Delta\theta_i$ and the sum is

$$\Delta D_a = \sum_{\Delta\theta_i>0} \Delta\theta_i \qquad \text{(A1)}$$

Also, assume that there are n_b points with negative $\Delta\theta_i$ and the sum is

$$\Delta D_b = \sum_{\Delta\theta_i<0} \Delta\theta_i \tag{A2}$$

Rescale all positive $\Delta\theta_i$ by

$$\Delta\theta_i' = \Delta\theta_i[1 - (\omega\, n_a \theta_{TE})/(n\Delta D_a)] = S_a\, \Delta\theta_i \tag{A3}$$

where $n = n_a + n_b$ and ω is an adjustable constant which ranges from one to two.

Rescale all negative $\Delta\theta_i$ by

$$\Delta\theta_i' = \Delta\theta_i[1 - (\omega\, n_b \theta_{TE})/(n\Delta D_b)] = S_b\, \Delta\theta_i \tag{A4}$$

Notice that the minimum magnitude of the rescaling factors S_a and S_b in Equations (A3) and (A4) should be bounded by a positive constant $S_l < 1$ to preserve the sign of Equations (7) and (11) upon correction. The maximum magnitude should also be bounded by another limiter $S_u > 1$ to prevent unreasonable local scaling. The effect of this $\Delta\theta_i$ rescaling is the reduction of the trailing edge mismatch and therefore the amount of rigid body rotation required to eliminate the mismatch.

There is some question as to when this procedure should be used. It is possible that for some ΔC_p distributions, the rescaling constants S_a and S_b in Equations (A3) and (A4) may become much smaller or much larger than unity. Consequently, Equations (7) and (11) can be drastically changed by rescaling. Setting the two limiters S_l and S_u close to one may lessen the change to an acceptable level. However, such a choice of limiters will restrict the ability for the rescaling to be effective in terms of reducing the rigid body rotation. Therefore, this procedure should only be used in the early stage of design iterations when direct application of rigid body rotation encounters convergence problems. The limiters should not be too close to unity for the effective reduction of rigid body rotation. The optimum choice of such limiters may be case dependent and may require numerical experimentation. For the two test cases presented in Section 4, geometry closure was achieved by rigid body rotation without using the $\Delta\theta_i$ rescaling procedure.

Appendix B

Data Smoothing

Smoothing was used to ensure a smooth aerodynamic surface in the chordwise and circumferential directions on the nacelle. Smoothing was also found useful in the target C_p preparation. Two types of smoothing procedures are presented in this appendix; the first type uses least-squares curve fitting (refer to sub-section 3.3.2) and the second type uses simple averaging. Two model problems are used to numerically evaluate the characteristics of the smoothing procedures. The desirable characteristcs include preservation of peak location, preservation of symmetry, and preservation of area under the curve. Asymptotic behavior of the smoothing procedures will also be discussed. For simplicity, only evenly spaced grids are discussed in this Appendix. However, the concepts are easily amenable to uneven grids.

B.1 Least-Squares Smoothing

In the first model problem, we take a set of points

$$\{x_i , y_i\} , \quad i = 1, 41$$

$$x_i = i - 1 , \quad i = 1, 41$$

$$y_i = 0 , \quad i = 1, 20 \quad \text{and} \quad i = 22, 41$$

$$y_i = 1 , \quad i = 21$$

Graphical representation of a part of this data set is given in Figure B.1 which has a unit spike at x=20. After one pass of smoothing, when the smoothing procedure described in sub-section 3.3.2 was used on Equation (14), the spike was lowered and shifted to the left by one grid point. In addition, the symmetry is lost. Seven consecutive points were used to obtain a least-squares fit to Equation (14)

$$y = c_1 + c_2 x + c_3 x^2 + c_4 x^3 \qquad (14)$$

Three points on either side of the center point determined the coefficient of the polynomial. That is, the center point itself did not contribute to the value of the coefficient. These coefficients were then used to compute a new value for the center point. This explains why the spike was drastically reduced immediately after the curve fitting. This smoothing procedure is referred to as Option-1.

The shifting of the peak value was eliminated when all seven points were used to determine the polynomial coefficients, Option-2 in Figure B.1. The least-squares fitting swept from left to right, and the most updated value for each point was used. This destroyed the symmetry of the curve.

The symmetry was retained by using old values during sweeping. All points were updated only after sweeping, Option-3 in Figure B.1.

Figure B.2 shows that even for highly spiky data, ten passes of smoothing are sufficient to arrive at a smooth curve while retaining the peak location and symmetry.

Using trapezoidal rule, the integrated area A can be computed from

$$A = (y_1 + y_{41})/2 + \sum_{i=2}^{40} y_i \qquad (B1)$$

Computed A equals one for the pre-smoothed data of this model problem. Least-squares smoothing preserves the integrated area under the curve so long as the curve is symmetric and the smoothing does not yield nonzero y_i at i = 2 or i = 40. After that, the integrated area deviates from unity but the magnitude of such deviation was found to be less than 0.001 within 20 passes of smoothing using Option-3.

B.2 Smoothing by Averaging

The following three-point averaging formula

$$(y_i)^{n+1} = (y_{i-1} + 2\,y_i + y_{i+1})^n / 4, \quad i = 2, 40 \tag{B2}$$

was studied. In this formula, superscript n denotes data smoothed n times. Thus n=0 would be the original (un-smoothed) data. This formula is very appealing because of its simplicity and low operation counts per point. There is a two order of magnitude reduction of operation counts per point compared with the least-squares smoothing using Option-3. However, smoothing with (B2) behaves differently than that of the least-squares type. For example, Equation (B2) does not allow for a square-root type of curve fit near the leading edge. In addition, Equation (B2) does not conserve area but can be modified to do so.

For the model problem, Figure B.3 shows that the unit spike was well smoothed after ten passes. The peak location and the symmetry pattern were both intact. The area under the curve was also preserved. Smoothing with Equation (B2) will be referred to as Option-4.

To examine whether or not the area is preserved unconditionally. The incremental area ΔA between successive smoothing is computed from

$$\Delta A = A^{n+1} - A^n \tag{B3}$$

where the superscript n for A denotes area computed after the nth pass of smoothing. From Eqs (B2, B3)

$$\Delta A = (y_1 + y_{41} - y_2 - y_{40})^n / 4 \tag{B4}$$

Therefore, the area will be preserved if $\Delta A = 0$ in Equation (B4), or

$$(y_1 + y_{41} - y_2 - y_{40})^n = 0 \tag{B5}$$

Model problem two provides a numerical illustration of how area changes due to smoothing. In this problem, the original data set represents a roof-top function (Fig. B.4). The initial area is

$$A^0 = 20$$

After ten passes of smoothing (Fig. B.4), the roof-top of the initial area has been smoothed out and some area is lost. The area reduces to

$$A^{10} = 19.75$$

That is, each pass of smoothing results in an area change of

$$\Delta A = - 0.025$$

For area conservation, the smoothing formula Equation (B2) must be modified next to the end point at i = 2 and i = 40 by

$$(y_2)^{n+1} = (0.75\, y_2 + 0.25\, y_3)^n$$

$$(y_{40})^{n+1} = (0.75\, y_{40} + 0.25\, y_{39})^n \qquad \text{(B6)}$$

With this modification, and using Equations (B2, B3) it can be shown that the area is conserved. This weighted averaging smoothing procedure will be referred to as Option-5. The special end point formula (B6) can lead to non-smoothness next to the end point. In Figure B.5 after one pass of smoothing, a slight kink shows up next to either side of the end point. The kink gets much stronger by the tenth pass of smoothing. The area lost near the roof-top has been offset by the area gained in the vicinity of the kink.

It is undesirable to have a smoothing procedure that may introduce additional kinks (non smoothness) into the data. Option-4 is chosen instead of option-5. The area change is quite small if only a small number of smoothing passes are applied. In addition, the real data usually has peaks and valleys and there may be a cancellation effect between area gain and area loss. Asymptotic behavior of the smoothing procedures is discussed below.

B.3 Asymptotic Behavior

When the smoothing formula is used successively and indefinitely, it is interesting to note the asymptotic behavior of the formula. Let

$$\Delta y_i = (y_i)^{n+1} - (y_i)^n \tag{B7}$$

When the iteration converges $\Delta y_i = 0$, or

$$(y_i)^{n+1} = (y_i)^n \tag{B8}$$

For option 4, from Equations (B2, B8)

$$(y_{i-1} - 2\, y_i + y_{i+1})^n = 0, \quad i = 2, 40 \tag{B9}$$

This condition can be satisfied if and only if the curve becomes a straight line connecting the two end points.

For option 5, from Equations (B2, B6, B8)

$$\begin{aligned} &(y_{i-1} - 2\, y_i + y_{i+1})^n = 0, \quad i = 3, 39 \\ &(y_2)^n = (y_3)^n \\ &(y_{40})^n = (y_{39})^n \end{aligned} \tag{B10}$$

The converged curve becomes a trapezoid whose area equals that of the pre-smoothed data. It is obvious that one should not attempt to converge the smoothing iteration. In practice, the data is not as spiky as that of the model problems, and a few passes (i. e., three to five) of smoothing are sufficient to arrive at a smoothed curve.

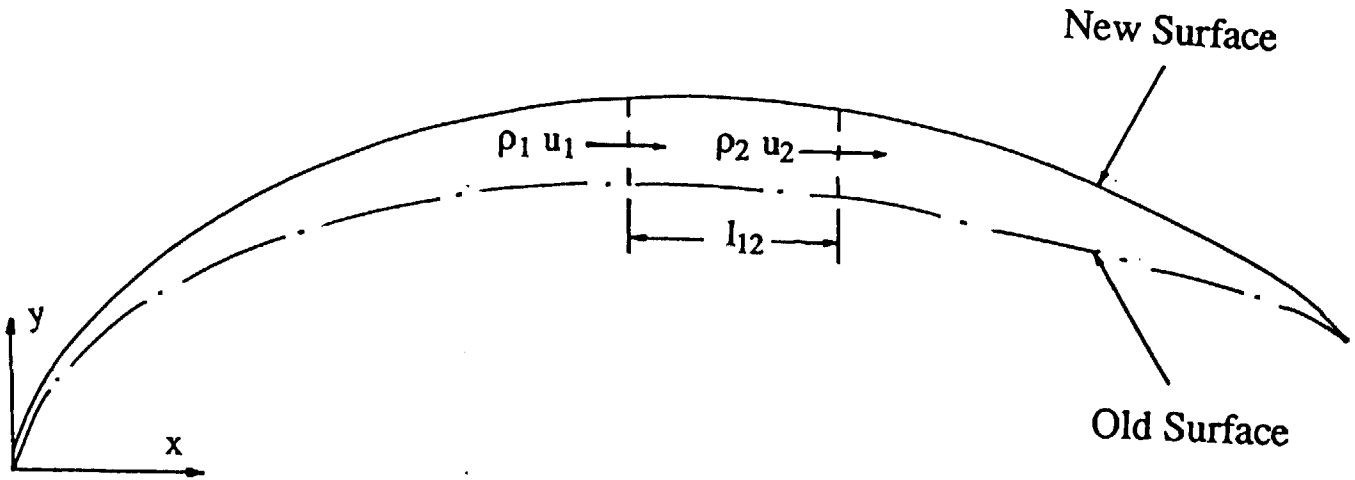
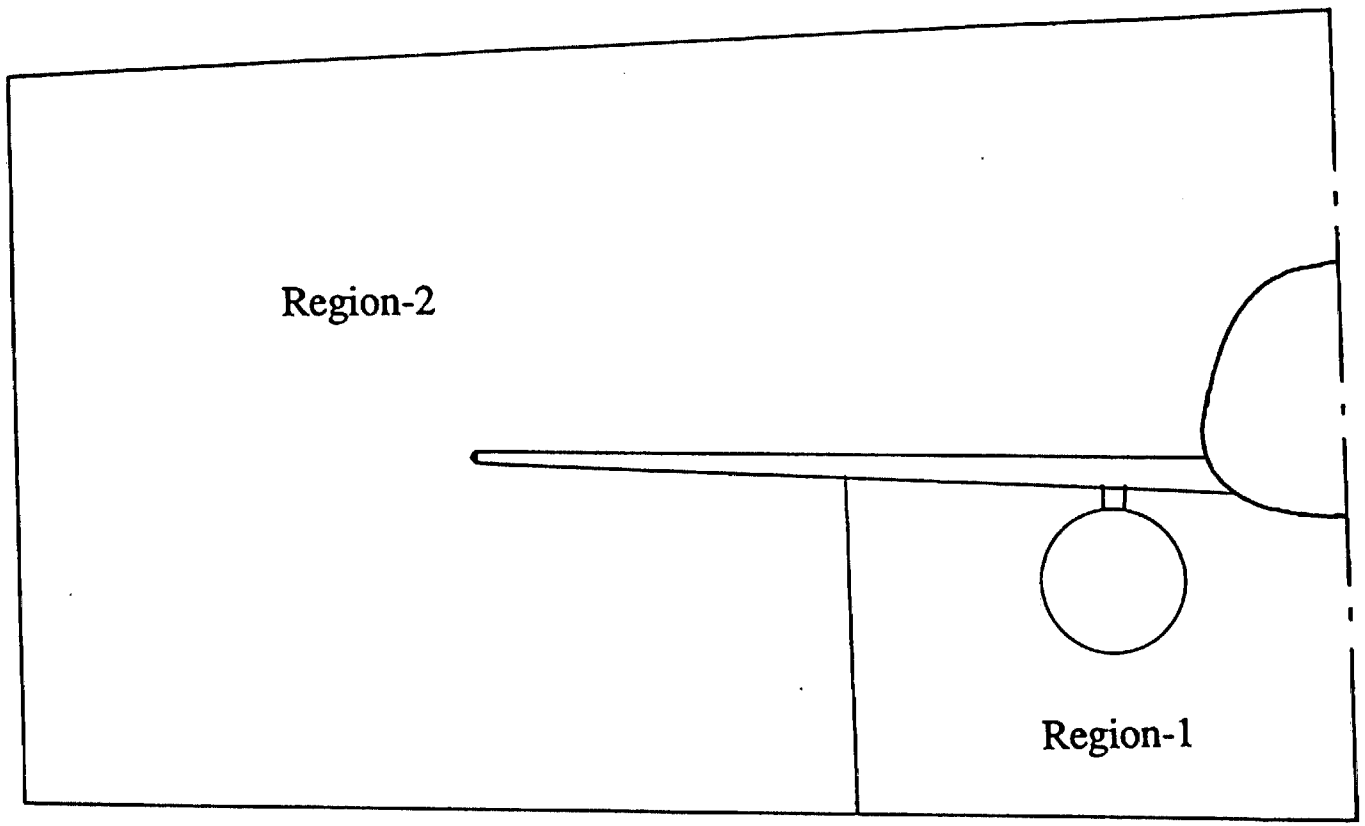
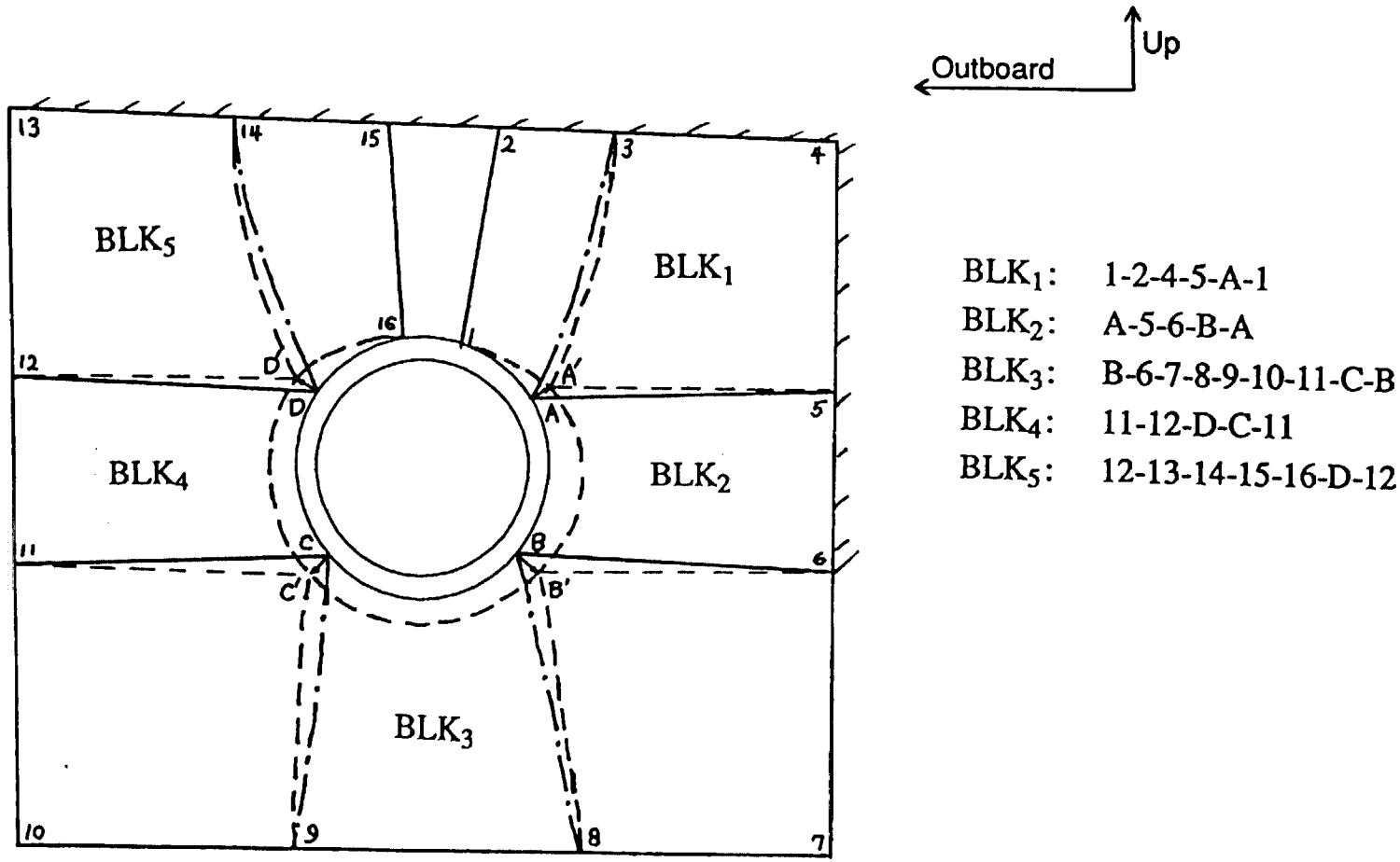


Figure 1. Flux Balance for Computing Surface Transpiration.



a) Flow Field Divided Into Two Regions

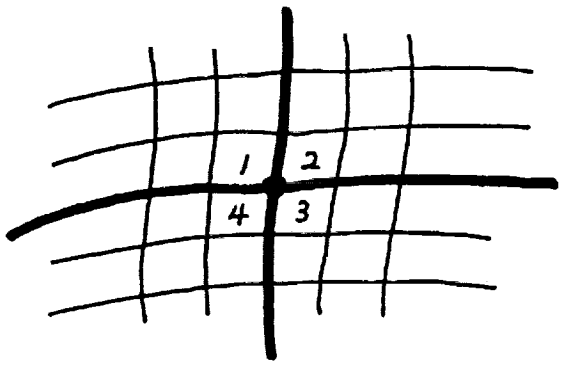
Figure 2. Block Layout for Re-gridding.



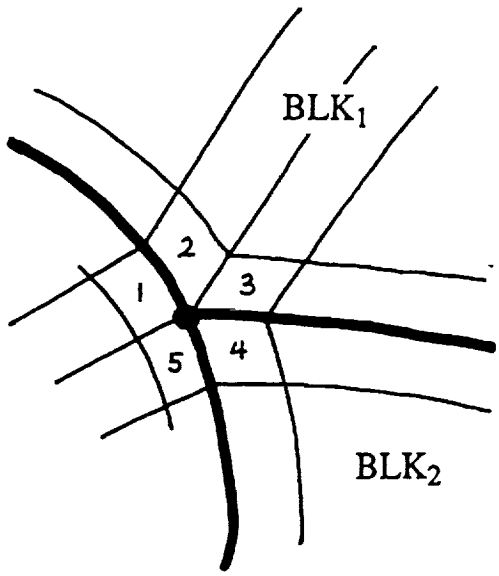
BLK$_1$: 1-2-4-5-A-1
BLK$_2$: A-5-6-B-A
BLK$_3$: B-6-7-8-9-10-11-C-B
BLK$_4$: 11-12-D-C-11
BLK$_5$: 12-13-14-15-16-D-12

b) Block Layout in Region-1

Figure 2. Block Layout for Re-gridding (Concluded).



a) Regular Point



b) Singular Point at Fictitious Edge

Figure 3. Regular and Singular Points at Block Boundaries.

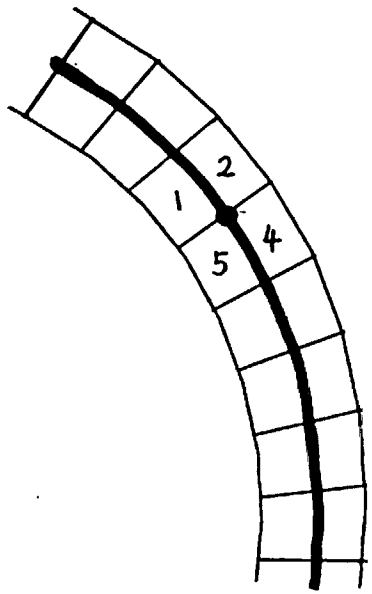


Figure 4. Data Reorganization into Circumferential Rings to Eliminate Singular Points

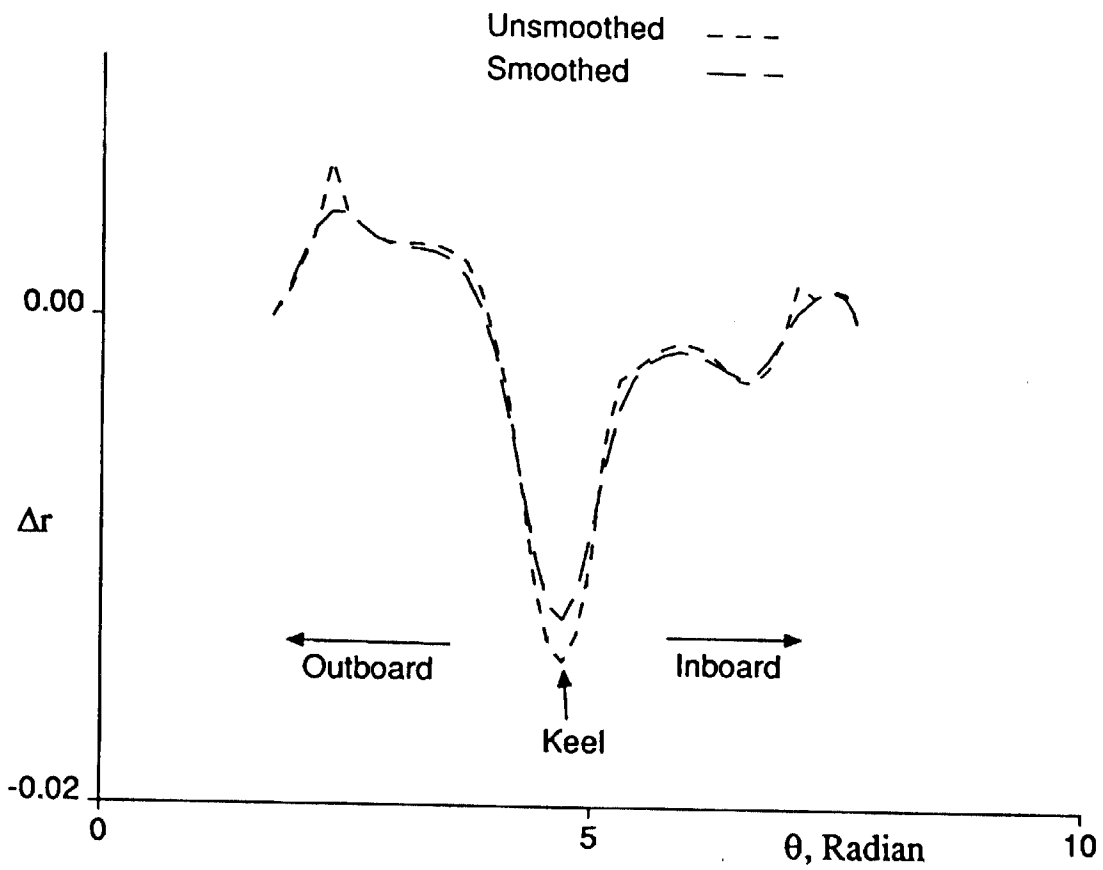


Figure 5. Circumferential Smoothing of Δr.

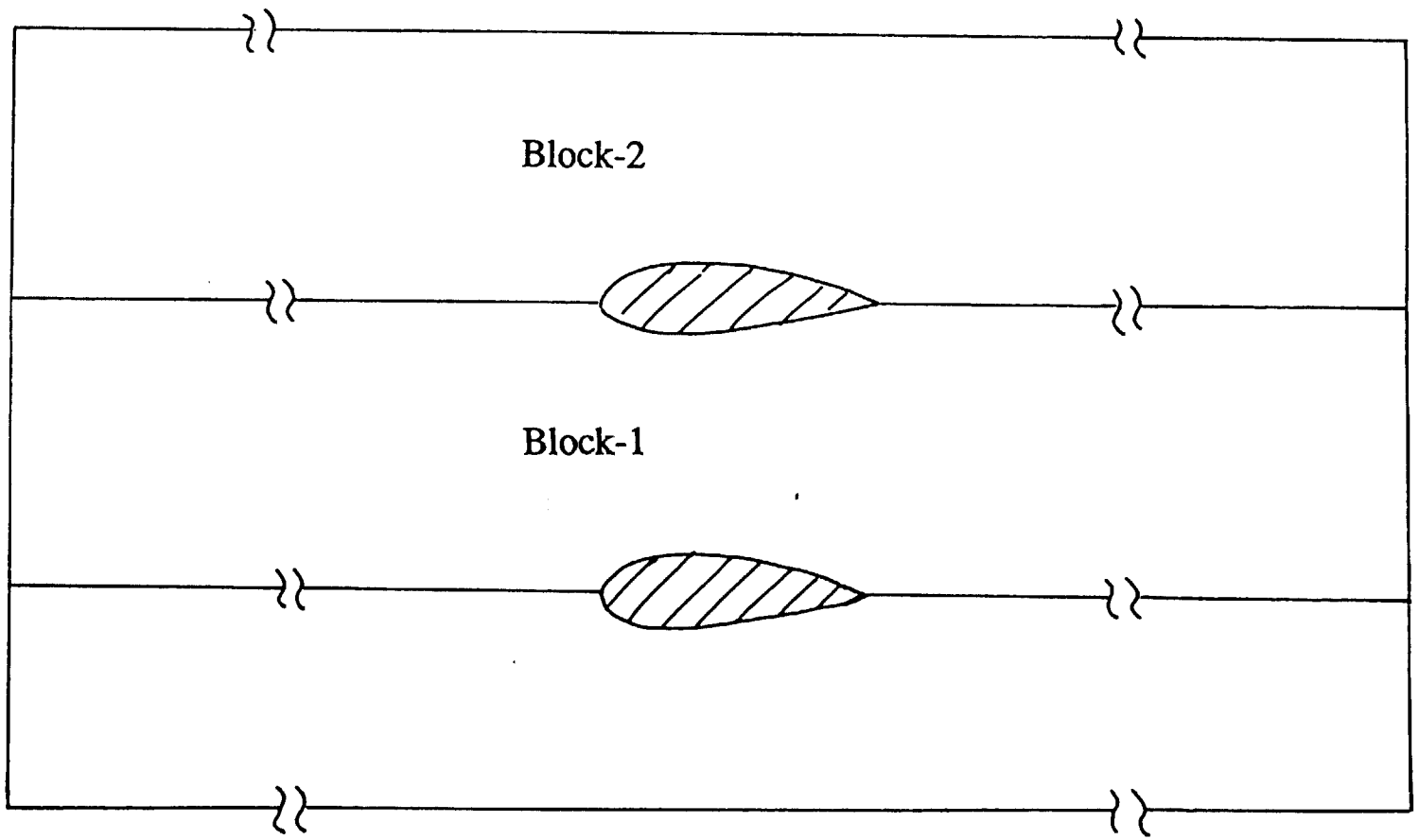


Figure 6. Block Layout for Isolated Nacelle.

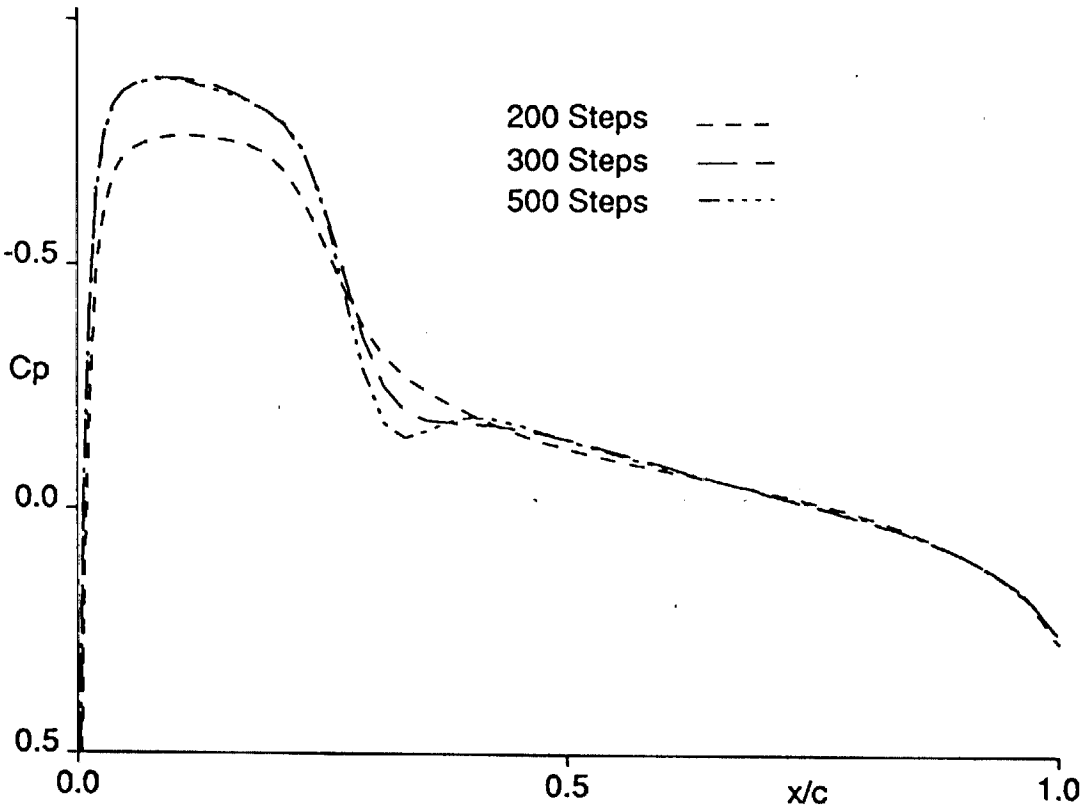


Figure 7. Convergence of C_p for an Isolated Nacelle, Nacelle-1, M_∞=.85, α=0 degrees.

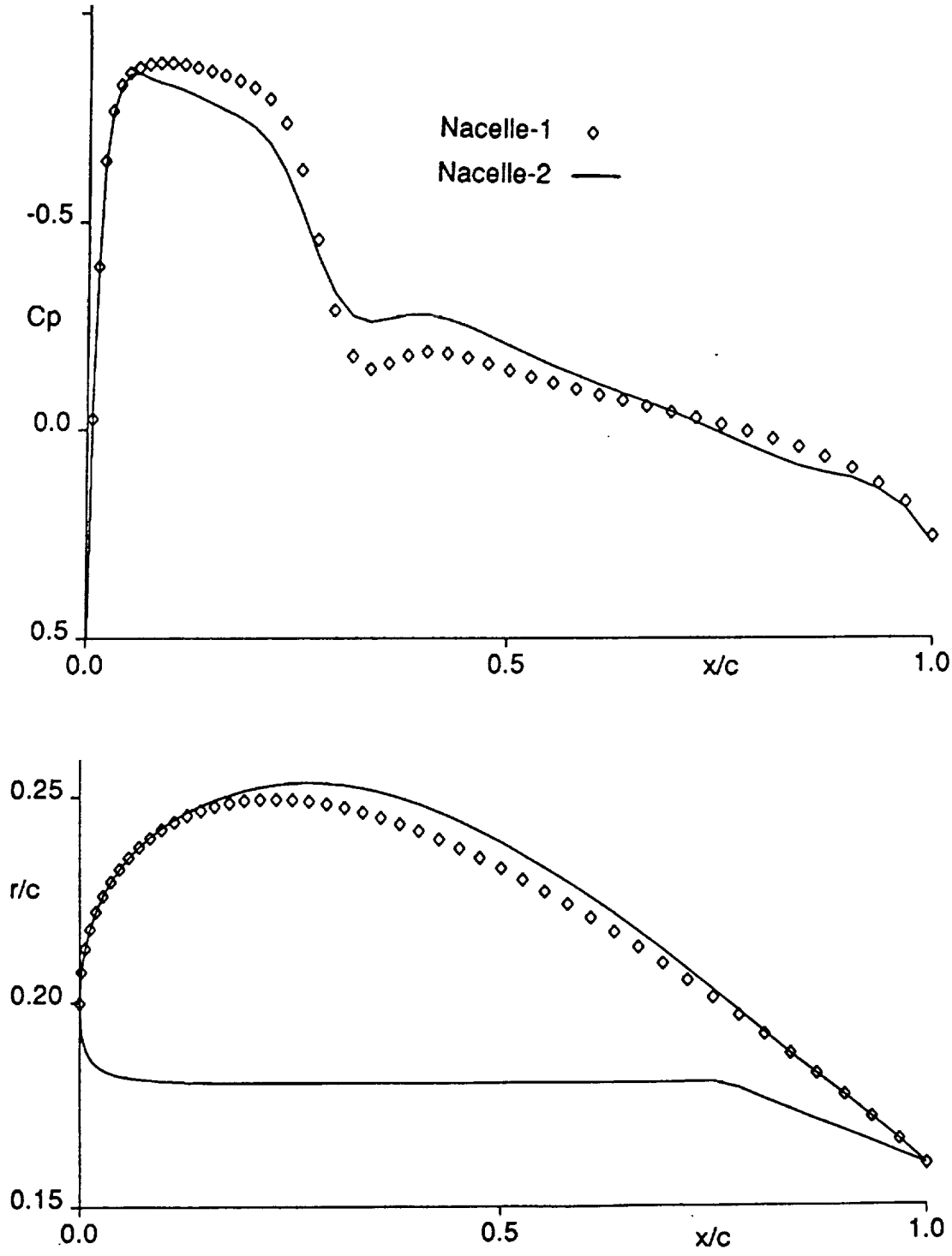


Figure 8. Comparison of C_p and Geometries Between Two Nacelles, M_∞=.85, α=0 degrees.

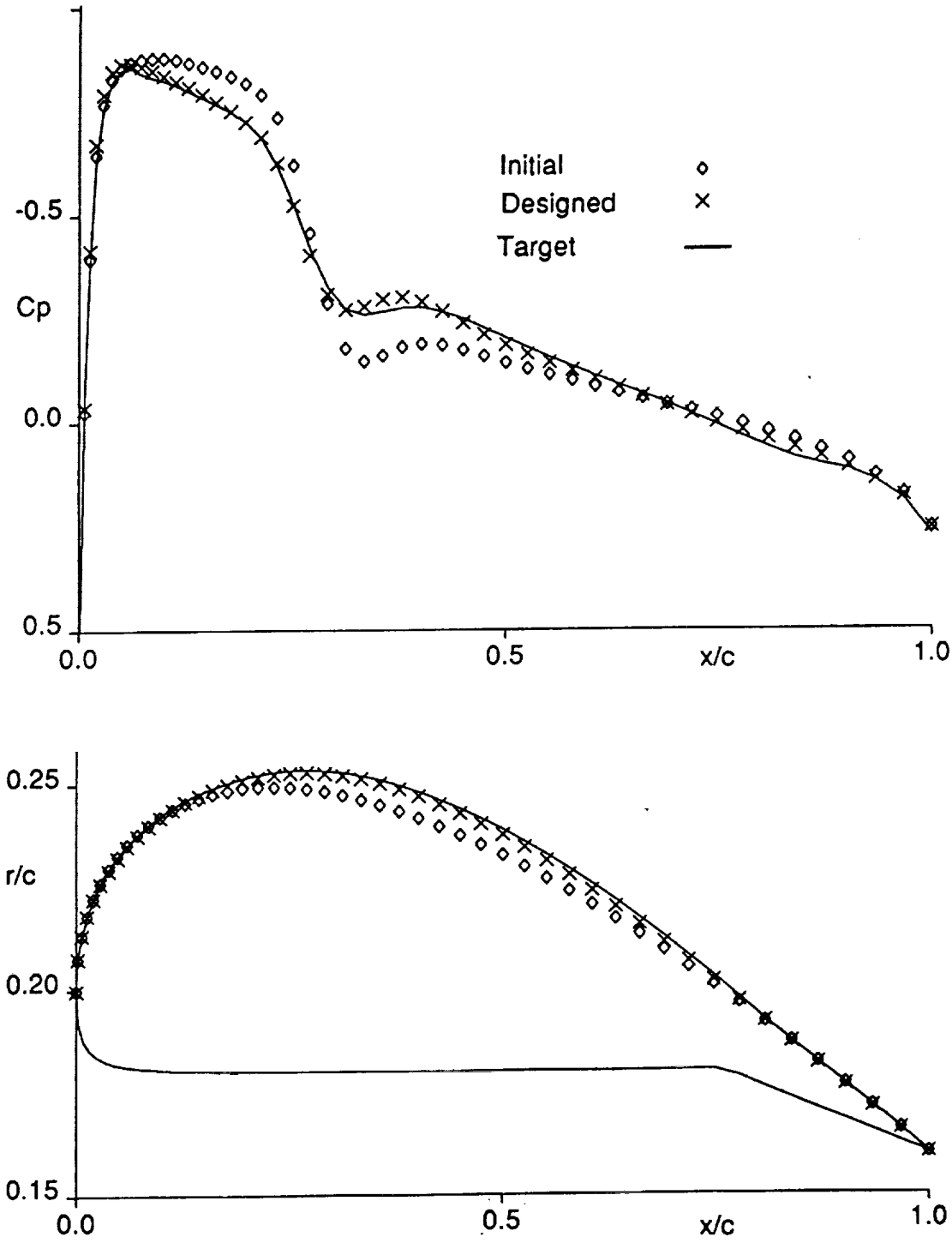


Figure 9. Design of an Isolated Nacelle, M_∞=.85, α=0 degrees.

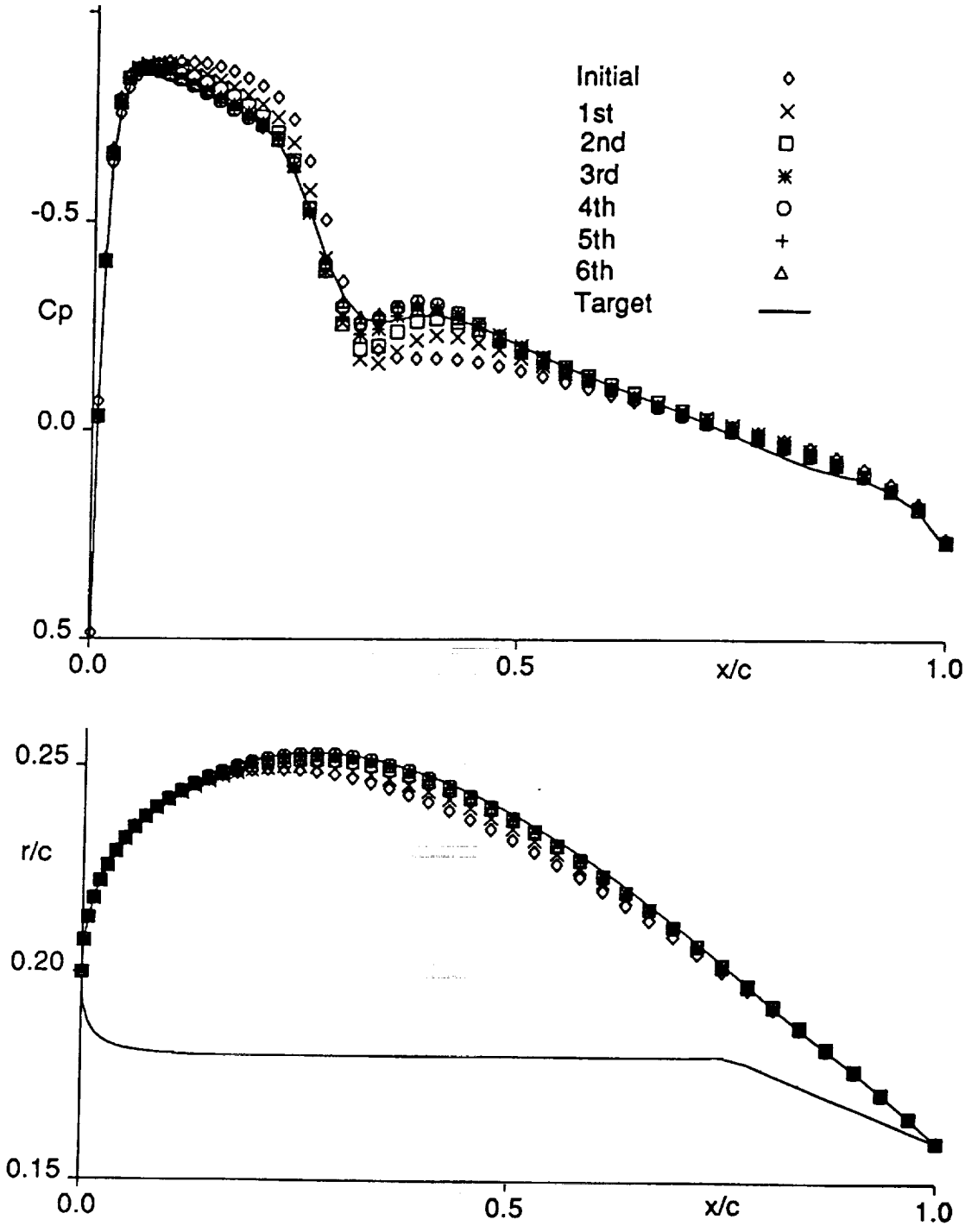


Figure 10. Convergence of Design Iterations for an Isolated Nacelle, M_∞=.85, α=0 degrees.

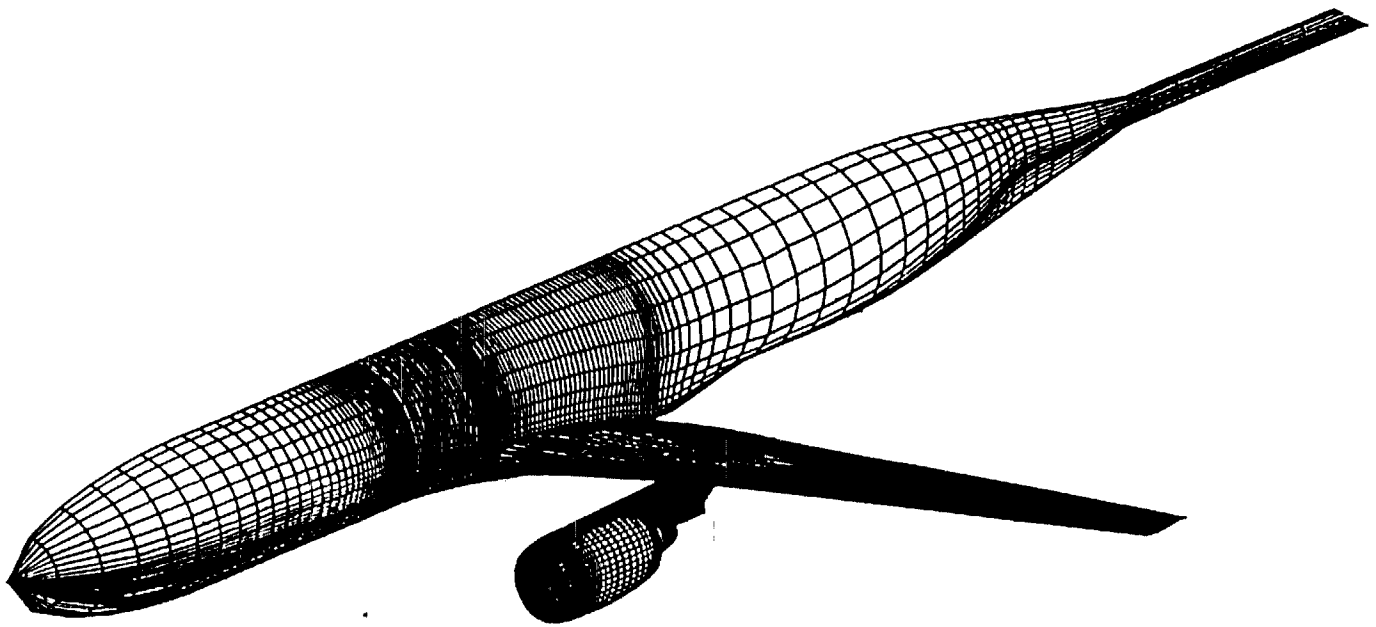


Figure 11. Surface Grid for an Engine-Pylon-Airframe Configuration.

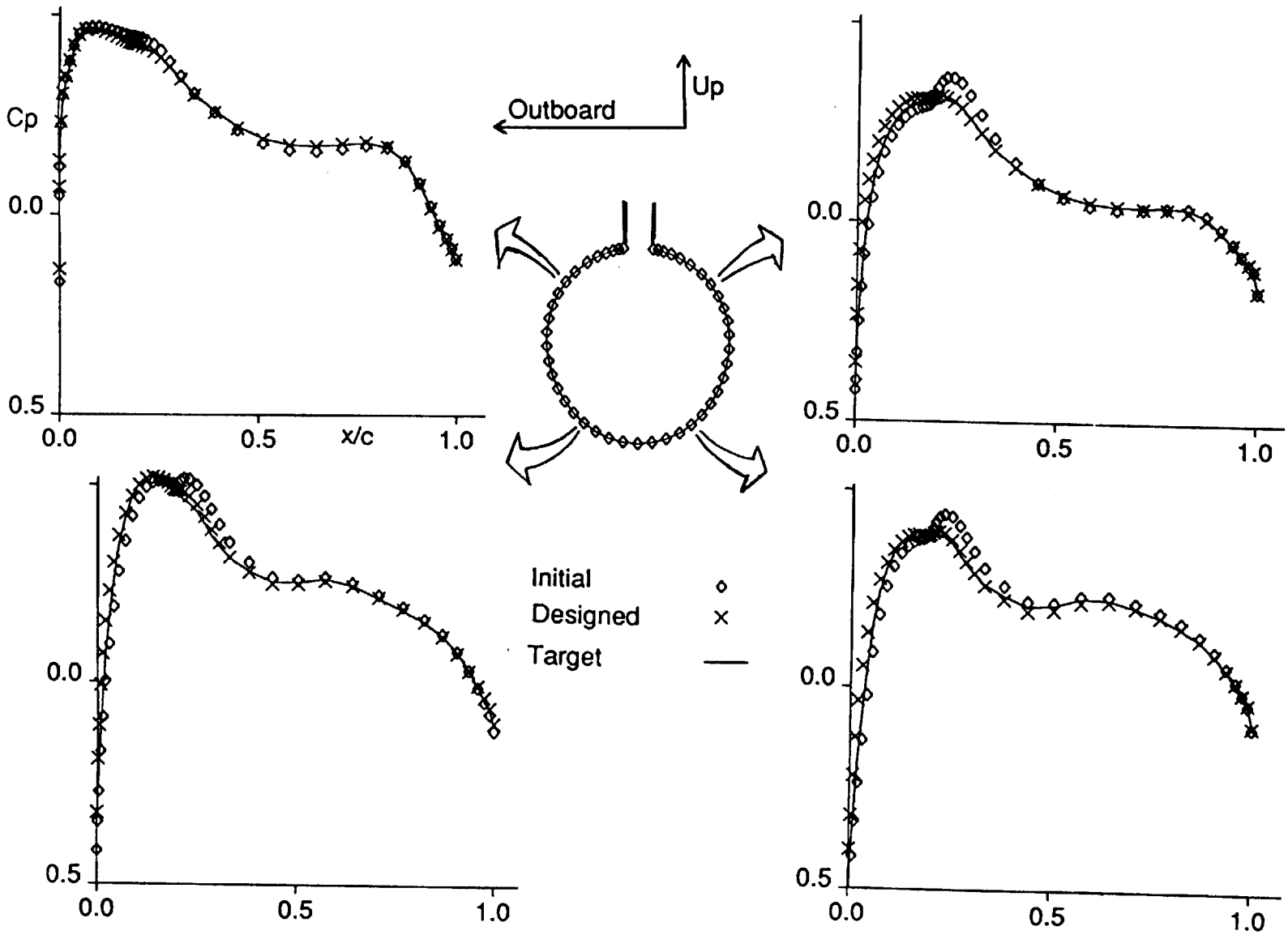


Figure 12. C_p Comparison in Installed Nacelle Design, M_∞=.77, α=.5 degrees.

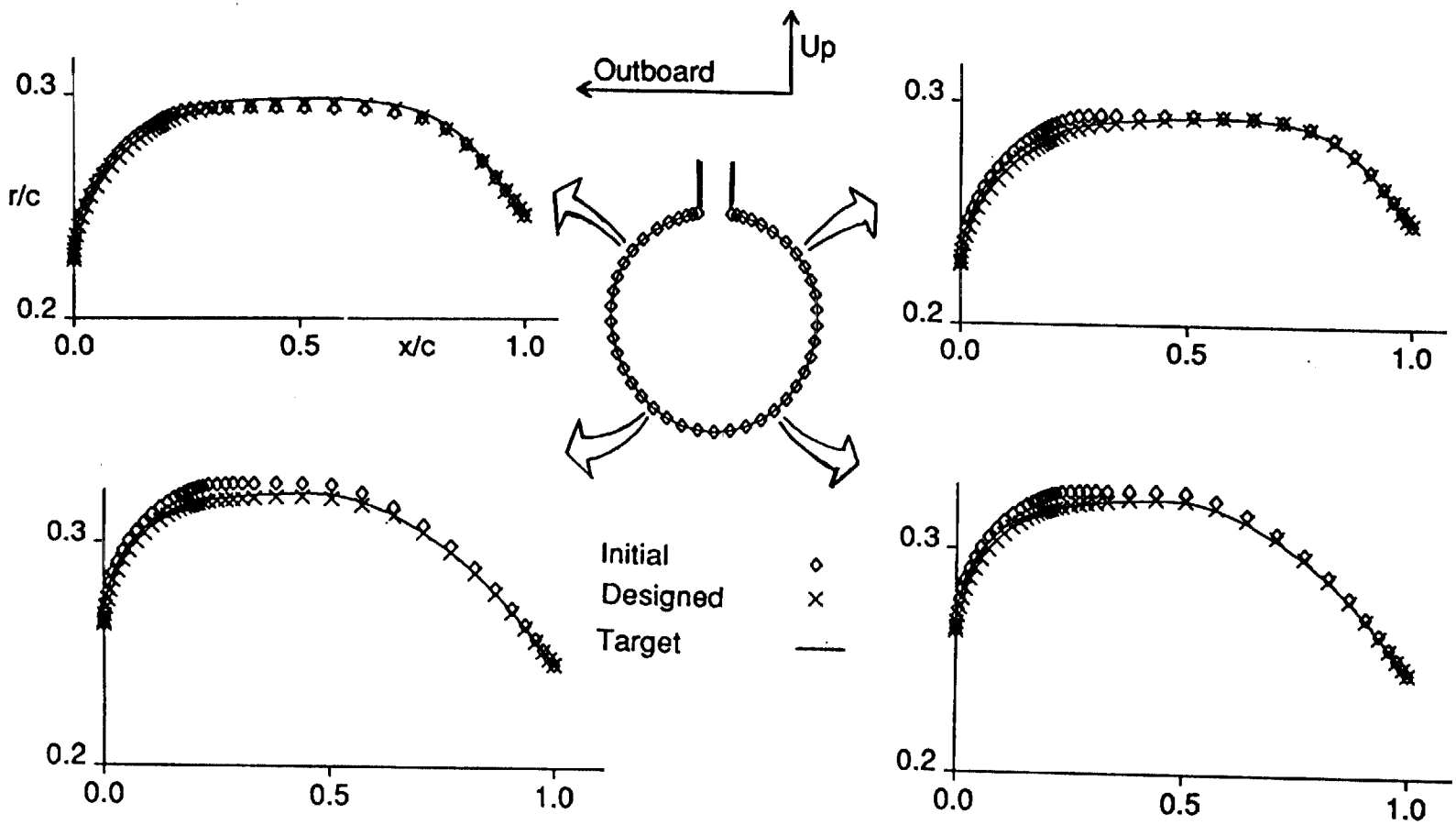


Figure 13. Recovery of Surface Geometry by the Design, M_∞=.77, α=.5 degrees.

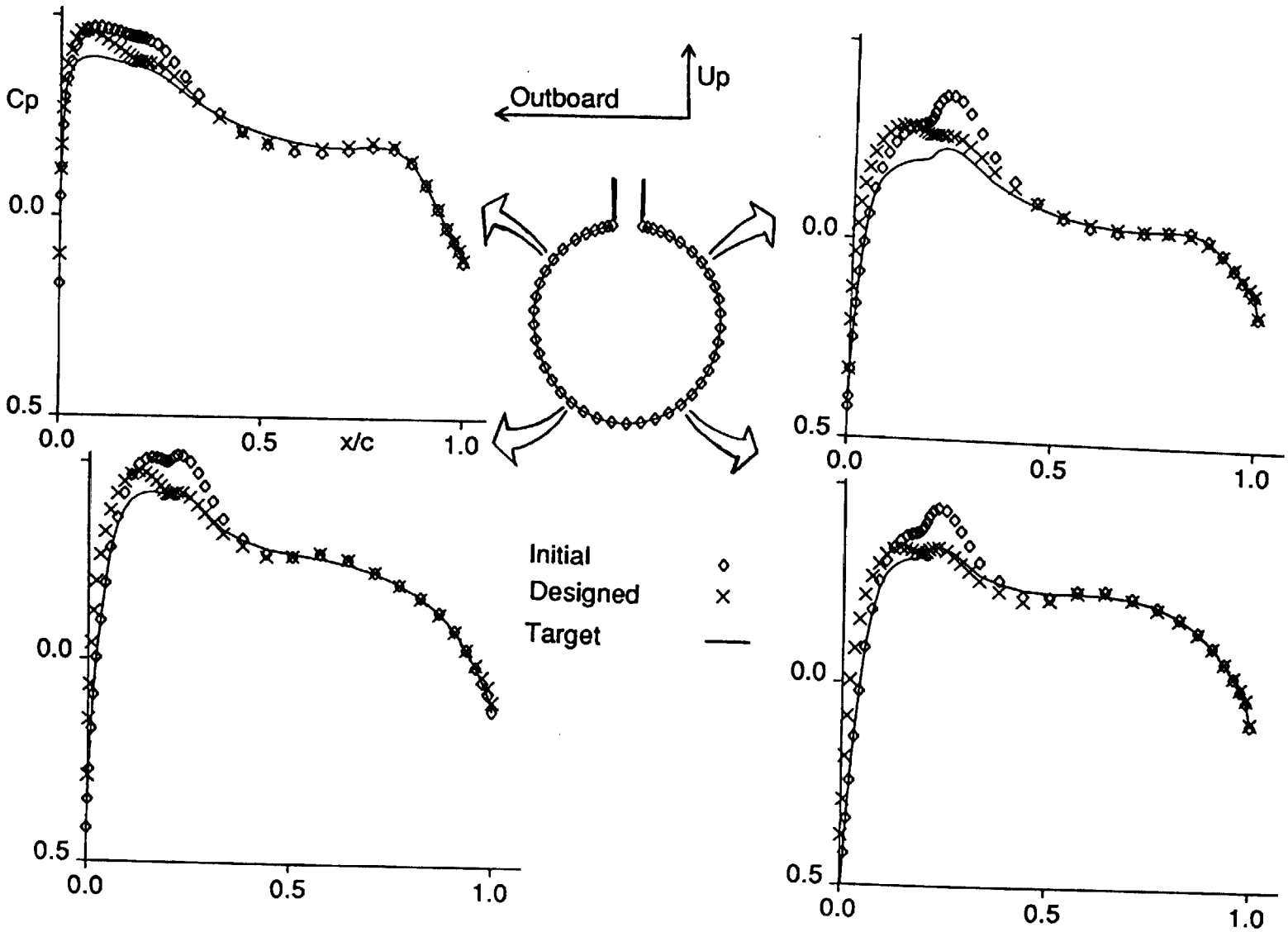


Figure 14. Installed Nacelle Design to Reduce the Pressure Peak, M_∞=.77, α=.5 degrees.

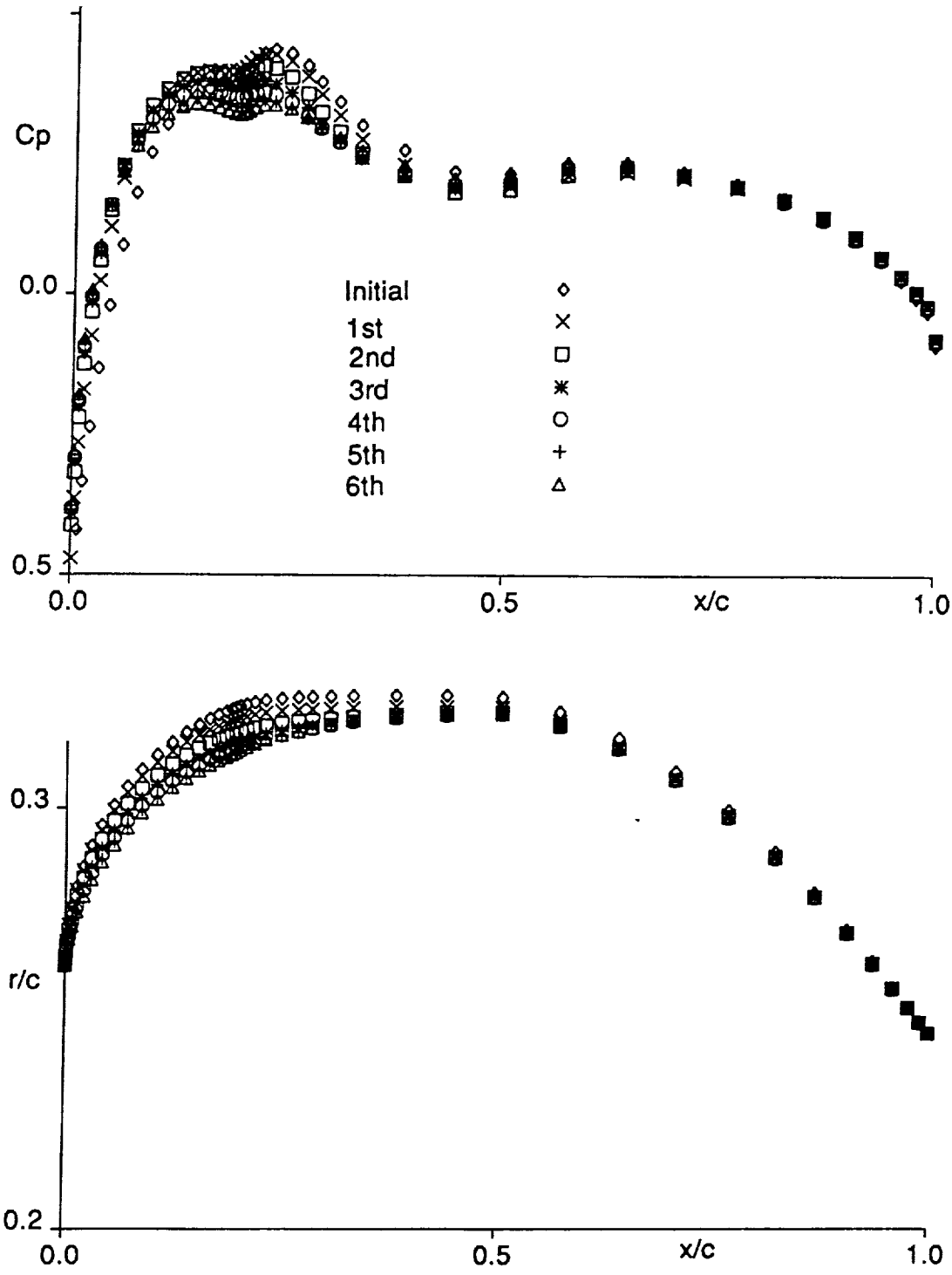


Figure 15. Convergence of C_p and Geometries in Installed Nacelle Design, M_∞=.77, α=.5 degrees.

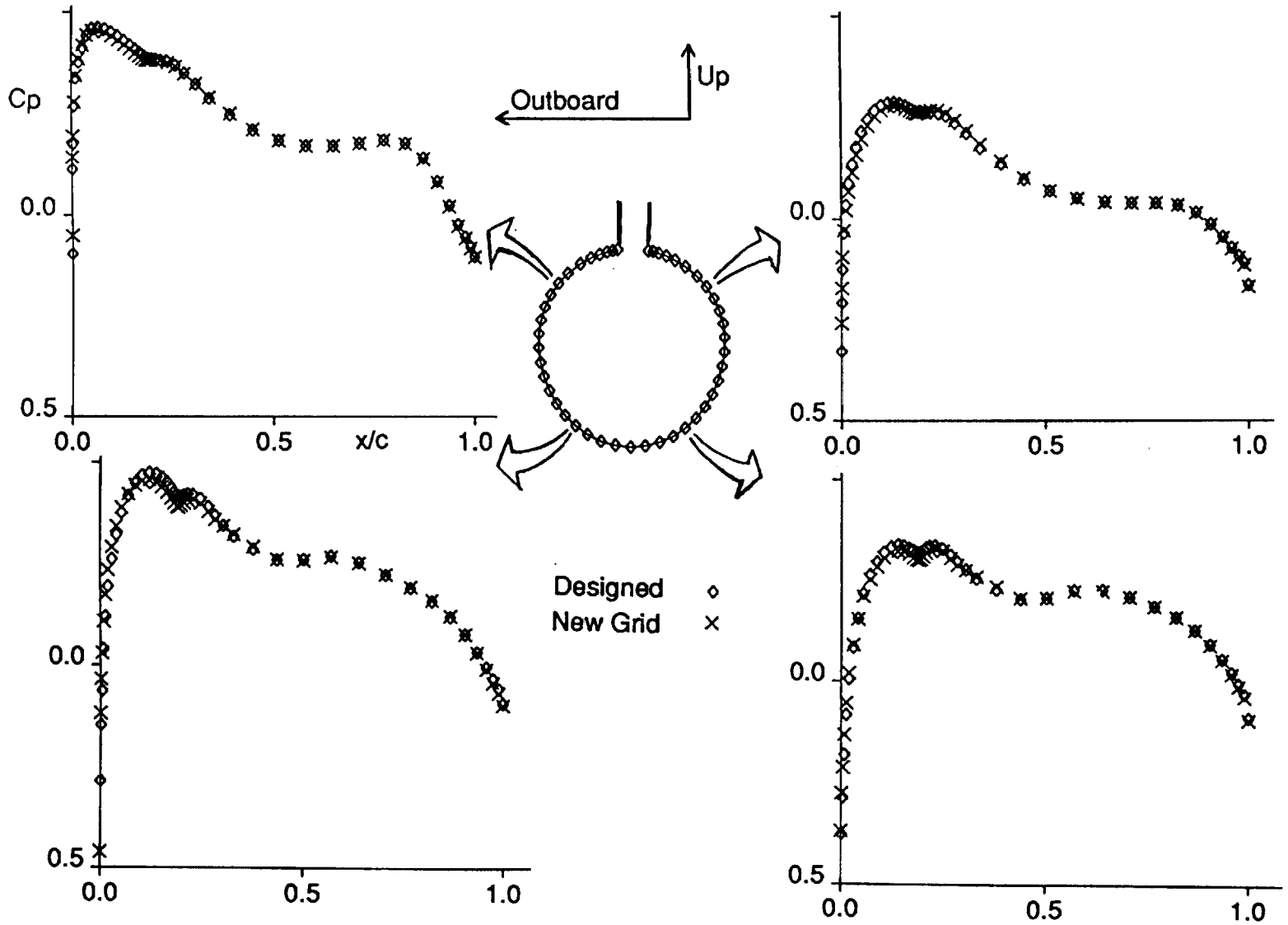


Figure 16. Verification of Design Using Transpiration Boundary Conditions, M_∞=.77, α=.5 degrees.

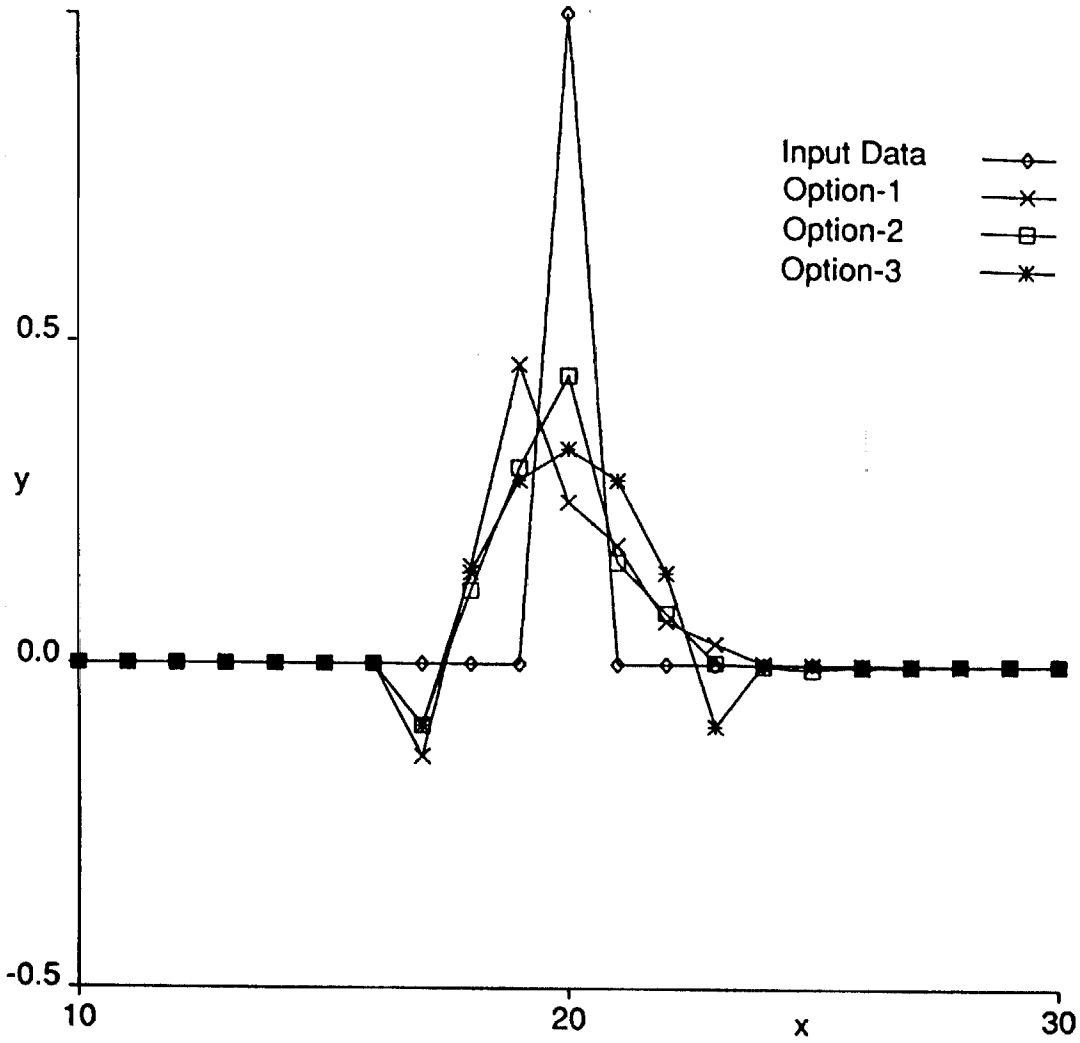


Figure B.1. Comparison of Three Least-Squares Smoothing Schemes After One Pass.

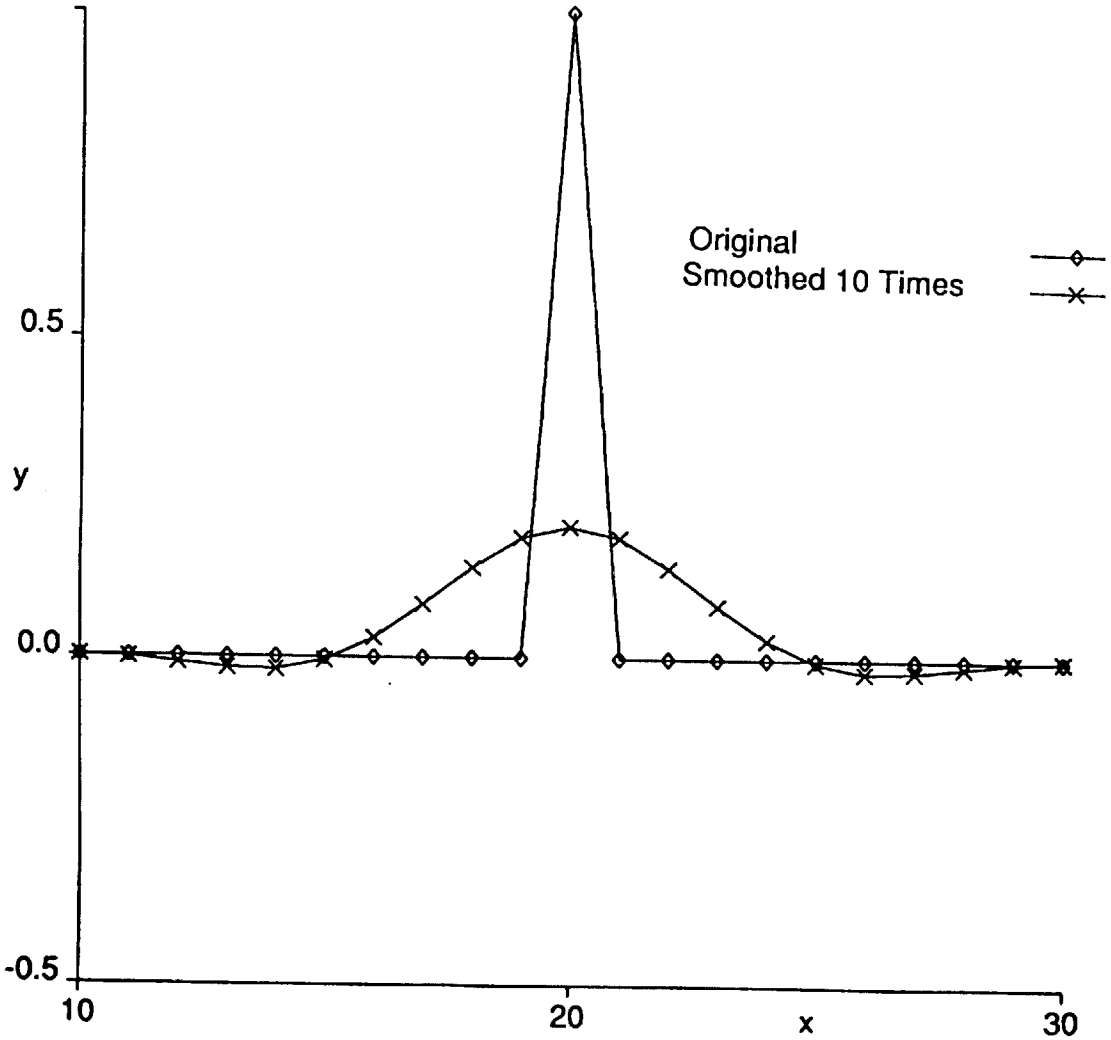


Figure B.2. Smoothing Option-3 (Least-Squares).

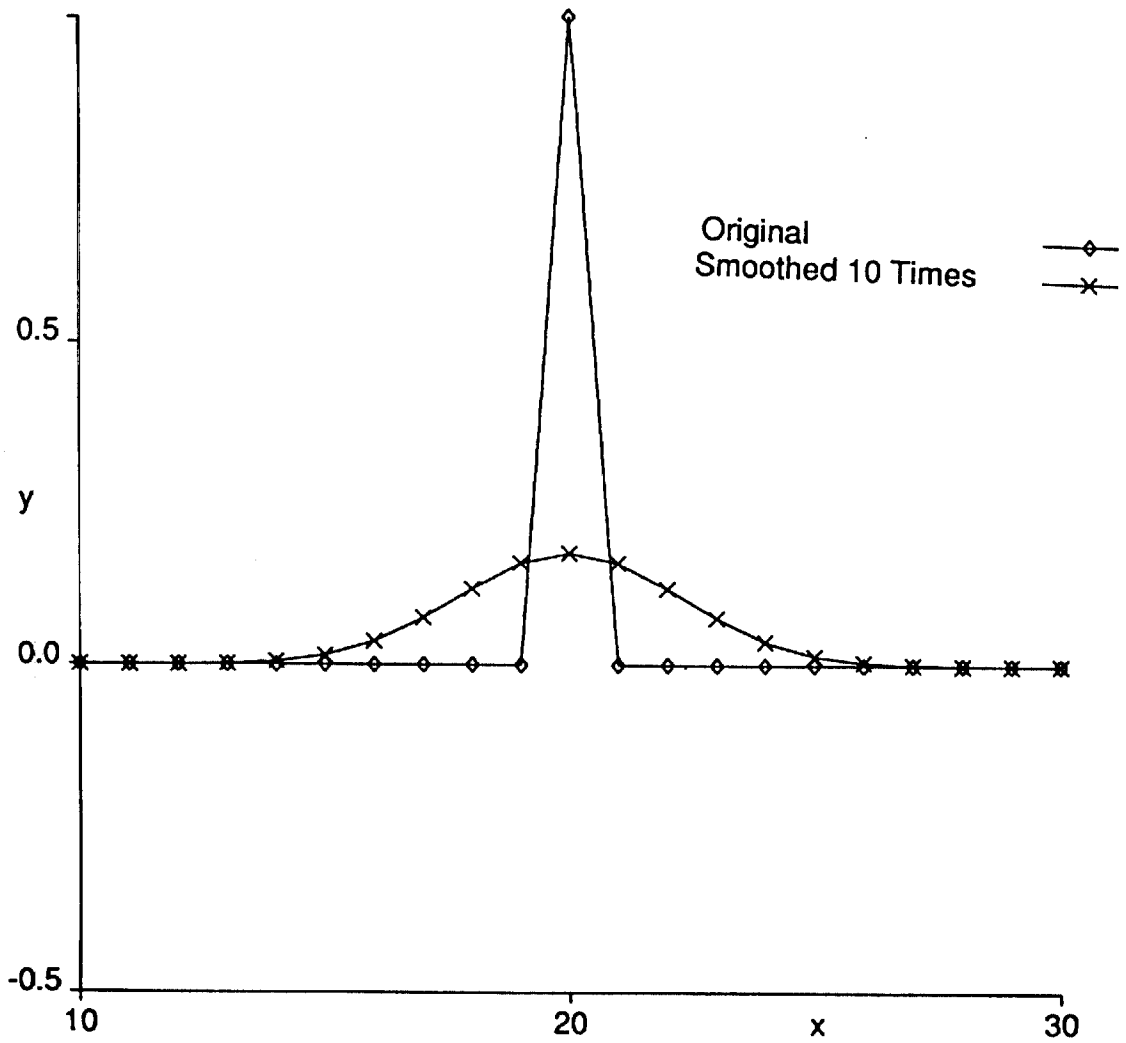


Figure B.3. Smoothing Option-4 (Averaging).

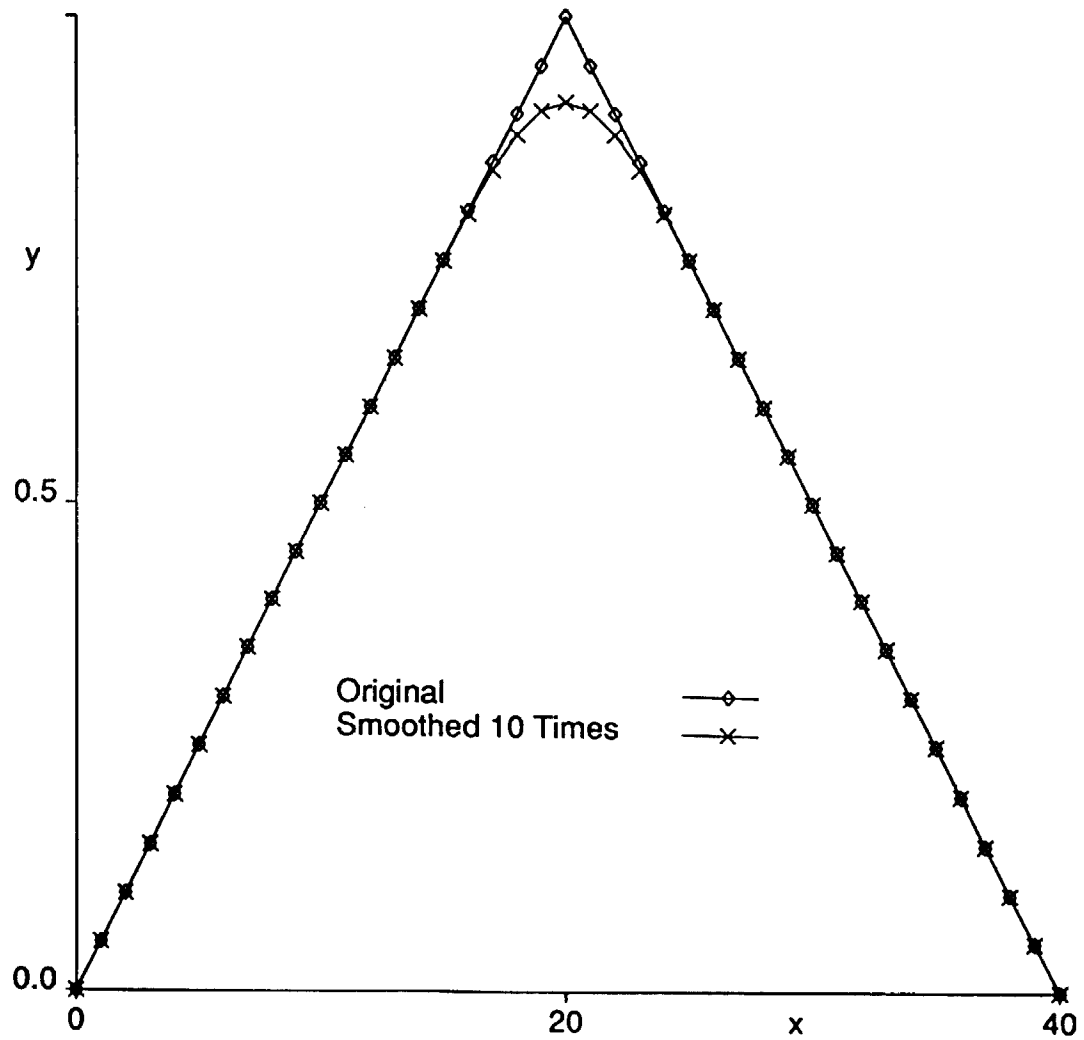


Figure B.4. Loss of Area When Smoothing by Option-4 (Averaging).

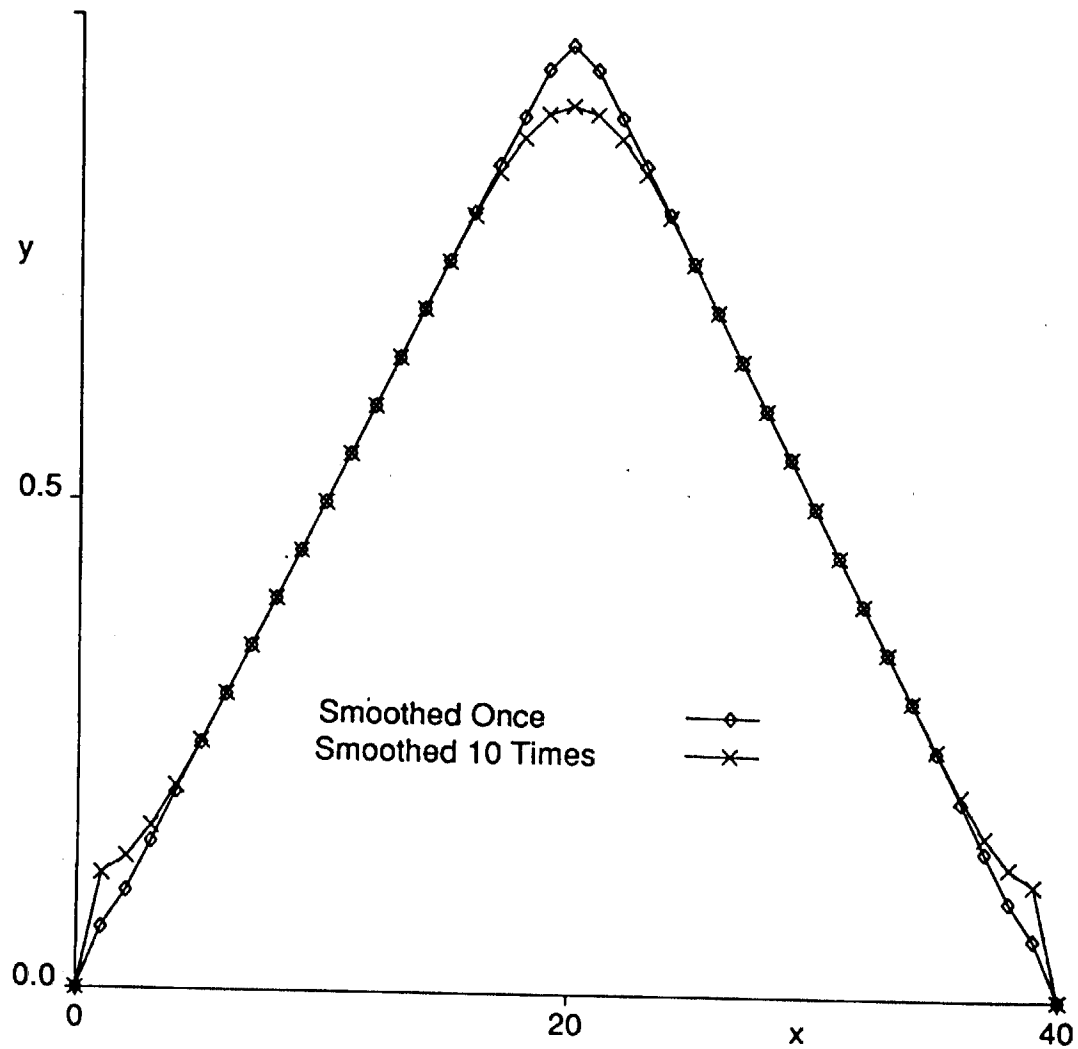


Figure B.5. Effect of Special Treatment Near the Ends (Option-5).

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	September 1992	Contractor Report

4. TITLE AND SUBTITLE
An Installed Nacelle Design Code Using a Multiblock Euler Solver, Volume I: Theory Document

5. FUNDING NUMBERS
NAS1-18703

6. AUTHOR(S)
H. C. Chen

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Boeing Commercial Airplane Group
P. O. Box 3707
Seattle, WA 98124-2207

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
National Aeronautics and Space Administration
Langley Research Center
Hampton, VA 23665-5225

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES
Langley Technical Monitor: Bobby Lee Berrier
Final Report

12a. DISTRIBUTION / AVAILABILITY STATEMENT
Unclassified Unlimited
Subject Category 02

12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)

An efficient multiblock Euler design code was developed for designing a nacelle installed on geometrically complex airplane configurations. This approach employed a design driver based on a direct iterative surface curvature method developed at NASA-Langley. A general multiblock Euler flow solver was used for computing flow around complex geometries. The flow solver used a finite-volume formulation with explicit time-stepping to solve the Euler equations. It used a multiblock version of the multigrid method to accelerate the convergence of the calculations. The design driver successively updated the surface geometry to reduce the difference between the computed and target pressure distributions. In the flow solver, the change in surface geometry was simulated by applying surface transpiration boundary conditions to avoid repeated grid generation during design iterations. Smoothness of the designed surface was ensured by alternate application of streamwise and circumferential smoothings. The capability and effiency of the code was demonstrated through the design of both an isolated nacelle and an installed nacelle at various flow conditions. Information on the execution of the computer program is provided in Volume II which is titled "An Installed Nacelle Design Code Using a Multiblock Euler Solver, Vol. II: User Guide".

14. SUBJECT TERMS	15. NUMBER OF PAGES
Installed Nacelle Design, Multiblock Euler Code	52
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified		